Cisco Data Center Assurance Program (DCAP) 4.0

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

Preface

The Data Center Assurance Program (DCAP) was created to provide a data center design solution that is tested persistently, completely, and objectively. This phase of the testing builds on the elements covered in the previous phase, and adds additional features and coverage. Future phases will repeat the testing executed in this phase as well as add testing for additional features and coverage. Testing is executed and results are reported as they were experienced. In short, the goal of DCAP is to provide transparency in testing so that our customers feel comfortable deploying these recommended designs.

About DCAP

The Data Center Assurance Program (DCAP) was created to provide a data center design solution that is tested persistently, completely, and objectively. This phase of the testing builds on the elements covered in the previous phase, and adds additional features and coverage. Future phases will repeat the testing executed in this phase as well as add testing for additional features and coverage. Testing is executed and results are reported as they were experienced. In short, the goal of DCAP is to provide transparency in testing so that our customers feel comfortable deploying these recommended designs.

The DCAP team does not exist as a standalone entity. Rather, it maintains close relationships with many successful teams within the Cisco testing community. The Enterprise Solutions Engineering (ESE) datacenter team supplies the starting point for datacenter topology design through its various SRND documents, which have been created through a close collaboration with marketing organizations and customer feedback sources. Testing direction is also guided by the Data Center Test Labs (DCTL) and Advanced Services (AS) teams, consisting of engineers who maintain tight relationships with customers while sustaining a solid track record of relevant and useful testing. Testing performed as part of Cisco DCAP 4.0 was undertaken by members of the Safe Harbor and NSITE test teams.

Table 1 lists ESE Data Center Design Guides that were referenced for this release. Where possible and sensible, these design guides are leveraged for the various technologies that will be implemented in DCAP. Visit http://www.cisco.com/go/srnd for more information on Cisco design guides.

Table 1 Relevant ESE Design Guides for DCAP 4.0

Design Guide	External URL
Data Center Infrastructure Design Guide 2.1	http://www.cisco.com/application/pdf/en/us/guest/netsol/ns107/c649/ccmigration_09186a008073377d.pdf
Data Center Infrastructure DG 2.1 Readme	http://www.cisco.com/application/pdf/en/us/guest/netsol/ns107/c133/ccmigration_09186a0080733855.pdf

Table 1 Relevant ESE Design Guides for DCAP 4.0 (continued)

Design Guide	External URL
Data Center Infrastructure DG 2.1 Release Notes	http://www.cisco.com/application/pdf/en/us/guest/netsol/ns107/c133/ccmigration_09186a00807337fc.pdf
Server Farm Security in the Business Ready Data Center Architecture v2.1	http://www.cisco.com/application/pdf/en/us/guest/netsol/ns376/c649/ccmigration_09186a008078e021.pdf
Enterprise Data Center Wide Area Application Services	http://www.cisco.com/application/pdf/en/us/guest/netsol/ns377/c649/ccmigration_09186a008081c7da.pdf
Data Center Blade Server Integration Guide	http://www.cisco.com/application/pdf/en/us/guest/netsol/ns304/c649/ccmigration_09186a00807ed7e1.pdf
Integrating Oracle E-Business Suite 11i in the Cisco Data Center	http://www.cisco.com/application/pdf/en/us/guest/netsol/ns50/c649/ccmigration_09186a00807688ce.pdf

There are other sources of design guidance as well that were leveraged in designing the DCAP 4.0 test environment, including white papers and implementation guides from third-party vendors. For a more robust list of resources used in DCAP 4.0, please see the Appendix.

The Safe Harbor testing team provides the starting point for DCAP software candidate selection through its proven methodology and code-hardening testing. Where applicable, each software image used in the DCAP test topology has been tested and passed, or is under test, by the Safe Harbor team in their own test topologies.

The key to the DCAP program is the customer involvement, whether direct or indirect. Customer interaction is maintained directly through DCAP team presence at forums such as Cisco Technical Advisory Board (TAB) conferences and through customer feedback through direct polling and conversations. Indirectly, the various customer account teams provide valuable insight into the data center-related issues that are concerning our customers and the direction that customers are moving as data center technologies evolve.

To help maintain this culture of customer feedback, the DCAP team invites the reader to subscribe to the following email aliases by sending an email with the subject “subscribe”:

- *safeharbor-dc-list@external.cisco.com* – provided for Cisco’s external customers interested in the DCAP program
- *safeharbor-release-info@cisco.com* – provided for Cisco sales engineers, CA engineers, account managers, or anyone with a customer that might benefit from DCAP testing

Additionally, there are a number of websites where DCAP program information can be found:

- http://www.cisco.com/go/dcap
- http://www.cisco.com/go/cvd
- http://www.cisco.com/go/datacenter
- http://www.cisco.com/go/srnd
- (Cisco Internal) http://wwwin.cisco.com/marketing/datacenter/programs/dcap.shtml
- (Cisco Internal) http://safeharbor.cisco.com/

The Cisco DCAP 4.0 Suite

Though all of the elements in the data center function as a whole, these elements can also be viewed individually. Cisco DCAP 4.0 testing was performed both on the individual technologies and on the data center as a whole. This Cisco DCAP 4.0 suite consists of an overview, 10 test volumes, an appendix, and 16 configuration volumes. Each test volume focuses on a particular component of the data center, with the final volume focusing on the data center as a whole. The appendix is used to document procedures and methods used in support of the testing, that may or may not be directly related to the testing itself.

Volume 1: Overview

This introductory chapter provides information on the testing methodology used in DCAP and a broad overview of the scope of this phase of testing. It also touches on hardware used from our 3rd party vendor partners such as NetApp, Hewlett-Packard and EMC. A summary of software used in this phase of testing is provided here.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 1: Overview" for respective testing information and details.

Volume 2: LAN (Layer 2-3) Infrastructure

The Cisco DCAP 4.0 LAN infrastructure is built around the Catalyst 6500 switching platform that provides for various features such as 10-Gigabit Ethernet connectivity, hardware switching, and distributed forwarding. While testing focuses on the Catalyst 6500 platform, the Catalyst 4948-10GE switch is also deployed to provide top-of-rack access to data center servers. The LAN infrastructure design is tested for both functionality and response to negative events.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 2: LAN (Layer 2-3) Infrastructure" for respective testing information and details.

Volume 3: LAN (Layer 4-7) Services

The modular Catalyst 6500 switching platform supports various line cards which provide services at Layers 4-7, such as the Content Switching Module (CSM), Firewall Services Module (FWSM) and Application Control Engine (ACE). The tests in this chapter focus on the ability of these Service Modules to work together to provide load-balancing, and security services to data center traffic.

Two physically different deployments were tested in Cisco DCAP 4.0. In one, the Aggregation Layer switches are used to house Service Modules and to provide aggregation for the Access Layer. In the other, the Service Modules are deployed in separate Service Chassis that are connected to the Aggregation Layer switches. Testing was performed on each of these physically different topologies.

The following two Service Module combinations were tested in the Aggregation Layer switch deployment.

- Content Switching Module (CSM), Firewall Services Module (FWSM), Secure Socket Layer Services Module (SSLSM), and Intrusion Detection Services Module (IDSM)
- Application Control Engine (ACE), FWSM, and IDSM)

Though the CSM, FWSM, SSLSM, and IDSM combination was set up in the DCa topology (integrated in the Aggregation Layer switch) for many of these tests, for the majority of Cisco DCAP 4.0 testing, the ACE, FWSM, and IDSM combination was used in the Aggregation Layer switch. In the Service Chassis deployment, only the CSM, FWSM, SSLSM, and IDSM combination was tested.

In all of the various hardware configurations that were used in the testing, the Network Application Module (NAM) was installed and configured, though it wasn't tested directly at any time.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 3: LAN (Layer 4-7) Services" for respective testing information and details.

Volume 4: Storage Area Networking (SAN)

The DCAP SAN topology incorporates Cisco MDS fabric director products and design guides, industry best practices, and storage vendor implementation guidelines to provide a SAN infrastructure that is representative of the typical enterprise data center environment. The centerpiece of the topology is the Cisco MDS 9513 multi protocol SAN director running SAN-OS version 3.1(3a). The Cisco MDS 9124e embedded SAN fabric switch is also part of the topology.

The topology provides redundant fibre channel connectivity for Linux and Windows hosts using QLogic and Emulex host bus adaptors (HBA) to three different types of fibre channel enterprise storage arrays, namely the EMC DMX3, NetApp FAS6070, and Hewlett Packard XP10000. The topology also provides redundant fibre channel connectivity for synchronous storage replication and fibre channel over IP (FCIP) connectivity for asynchronous storage replication. Delay simulators and cable spools allow modeling of a redundant data center environment for disaster recovery and business continuance testing. The topology is designed to use actual hosts and applications to generate test traffic to model actual customer environments as close as possible.

The topology also includes a Quantum (formerly ADIC) i500 Scalar tape library with two IBM LTO3 tape drives.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 4: Storage Area Networking (SAN)" for respective testing information and details.

Volume 5: Wide Area Application Services (WAAS)

Cisco Wide Area Application Services (WAAS) is an application acceleration and WAN optimization solution for geographically separated sites that improves the performance of any TCP-based application operating across a wide area network (WAN) environment. With Cisco WAAS, enterprises can consolidate costly branch office servers and storage into centrally managed data centers, while still offering LAN-like service levels for remote users.

The DCAP WAAS topology incorporates the Wide-area Application Engines (WAE) at the remote branch and in the data center, either at the DC WAN edge or at the aggregation layer. For TCP traffic redirection at the WAN edge of Data Center B, Web Cache Communication Protocol version 2 (WCCPv2) was used. At Data Center A the Cisco Application Control Engine (ACE) was used at the data center aggregation layer for transparent TCP redirection. The tests in this chapter focus on the functionality of the WAAS software on the WAE devices as well as the ability of the data center ACE and WAN Edge routers to intercept and redirect TCP-based traffic. Microsoft Exchange 2003 and Oracle 11i E-Business Suite traffic was sent and optimization and functionality were verified and quantified.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 5: Wide Area Application Services (WAAS)" for respective testing information and details.

Volume 6: Global Site Selector (GSS)

The Global Site Selector (GSS) leverages DNS's distributed services in order to provide high availability to existing data center deployments by incorporating features above and beyond today's DNS services.

The GSS devices are integrated into the existing DCAP topology along with BIND Name Servers and tested using various DNS rules configured on the GSS. Throughout the testing, the GSS receives DNS queries sourced from client machines as well as via DNS proxies (D-Proxies). The Name Server zone files on the D-Proxies are configured to nsfoward DNS queries to the GSS in order to obtain authoritative responses. Time-To-Live (TTL) values associated with the various DNS resource records are observed and taken into consideration throughout the testing.

The tests in this chapter focus on the fundamental ability of the GSS working together with existing BIND Name Servers in order to provide global server load balancing.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 6: Global Site Selector (GSS)" for respective testing information and details.

Volume 7: Bladeswitching

The HP c-Class BladeSystem is a complete infrastructure of servers, network management and storage integrated in a modular design, built to deliver the services vital to a business data center. By consolidating these services into a single enclosure, power, cooling, physical space, management, server provisioning and connectivity savings can all be benefited.

In the DCAP topology both the Intel-based BL460c and AMD-based BL465c were provisioned to run the front end Oracle 11i E-Business Suite web application. BL685c servers were provisioned to provide back-end database service with Oracle Real Application Clusters (RAC). VMware ESX 3.0.2 was installed on BL485c servers, which were set up with boot from SAN and clustered to provide VMotioning capabilities. Each ESX server hosted Oracle Web application, Exchange Server 2003 hosts, and Windows Server 2003 domain controllers. The integrated Cisco 3020 Layer 2+ switch provided network connectivity to the data center Aggregation Layer in Data Center A. Four switches were housed in the DCA blade chassis and each one was configured with a dual-port Etherchannel dual homed to the Aggregation Layer switches. The Blade Enclosure in Data Center B was deployed with pass-thru modules allowing each server to connect directly into the Access Layer Catalyst 4948 and 6500 switches. The tests in this chapter focus on the basic feature functionality of the 3020 switch and its response to negative events.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 7: Bladeswitching" for respective testing information and details.

Volume 8: Applications: Oracle E-Business Suite

This phase of Oracle application testing consisted of Oracle 11i E-business Suite (11.5.10.2) with Oracle Database (10gR2) on Real Application Clusters (RAC) in Active/Active Hybrid mode implemented across two active data centers. A single Oracle Application Tier was shared across two data centers making it Active/Active while Database Tier is Active in only one data center with data being replicated synchronously to the second Data center making it Active/Passive. The architecture deployed validates various Cisco products (including GSS, ACE, CSM and MDS) which made up the entire solution. Cisco WAAS technologies were leveraged to optimize Oracle Application traffic sent from branch offices.

The Oracle Vision Environment was leveraged for Application testing which includes generating real application traffic using the HP-Mercury Load Runner tool. Traffic generated was sent to both data centers from clients located at three branch offices. Tests include verifying the configuration and

functionality of E-business application integration with GSS, ACE, CSM, Active/Active hybrid mode and WAAS optimizations. Tests also cover failover and failback of E-business Application in data center disaster recovery situation.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 8: Applications: Oracle E-Business Suite" for respective testing information and details.

Volume 9: Applications: Microsoft Exchange

The Microsoft Exchange 2003 topology consisted of two Windows 2003 active/passive back end clusters, one in each data center. The primary cluster hosted the Exchange Virtual Server and the other cluster acted as a disaster recovery/business continuance standby cluster. The clusters use fibre channel to attach to storage from EMC, HP, and NetApp. This storage was replicated synchronously from the primary to the standby cluster. Tests included running Microsoft Jetstress on the primary cluster, failing the primary cluster over to the standby cluster, and failing the standby cluster back to the primary cluster. Client access for failover/failback testing was from Outlook 2003 clients at three remote branches via the MAPI protocol over the test intranet, which was accelerated by WAAS.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 9: Applications: Microsoft Exchange" for respective testing information and details.

Volume 10: Applications: TIBCO Rendezvous

TIBCO Rendezvous (RV) is a multicast-based messaging middleware of particular interest to those financial customers with trading floors as part of their business. TIBCO RV takes financial data feeds in and sends them out to interested receivers subscribed to various multicast groups. The tests in this chapter, performed against TIBCO RV v7.5, verify the functionality of the networking infrastructure in its ability to deliver these messages as well as validating the ability of the network infrastructure to deliver inter-DC multicast data.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 10: Applications: TIBCO Rendezvous" for respective testing information and details.

Volume 11: Data Center High Availability

Cisco DCAP 4.0 testing included disaster recovery testing for the Oracle 11i E-Business Suite, Oracle 10gR2 database, and Microsoft Exchange 2003 application test beds described above. The data center disaster recovery tests included failing both applications over to DCb, and then failing the applications back to DCa. Replication of SAN data over fibre channel (with write acceleration enabled) and replication of NAS data over IP (with WAAS optimization) were key enablers.

Failover testing started with a simulation of a disaster by severing all WAN and SAN links to and from DCa. Failback testing started with a controlled shutdown of applications in DCb. Application data created or modified in DCb during failover was replicated back to DCa as part of the failback procedure. Parts of the failover and failback procedures were automated with GSS, ACE, and CSM, and other parts were manual. For each test, a timeline of automatic and manual steps was constructed and two key metrics, the Recovery Point Objective (RPO) and Recovery Time Objective (RTO), were determined and reported.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 11: Data Center High Availability" for respective testing information and details.

Volume 12: DCAP Appendix

Cisco DCAP 4.0 appendices summarizes the configuration, deployment, and/or implementation details for SAN, Cisco GSS, WAAS, HP Blade Server, Oracle 11i and MS Exchange applications, high availability, Bill of Materials and Power Draw requirements, DCAP 4.0 Resource considerations, and a test list of the latest Safe Harbor certified software releases used in DCAP testing.

Refer to the associated Cisco Data Center Assurance Program (DCAP) 4.0 document, "Volume 12: DCAP Appendix" for details.

Volume 13: LAN (Layer 2-3) CSM Configurations

Layer 2-3 configurations are provided for testing considerations.

Refer to the Volume 2: LAN (Layer 2-3) Infrastructure, page -xxxvii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 14: LAN (Layer 2-3) ACE Configurations

Layer 2-3 configurations are provided for testing considerations.

Refer to the Volume 2: LAN (Layer 2-3) Infrastructure, page -xxxvii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 15: LAN (Layer 4-7) CSM Configurations

Layer 4-7 configurations are provided for testing considerations.

Refer to the Volume 3: LAN (Layer 4-7) Services, page -xxxvii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 16: LAN (Layer 4-7) ACE Configurations

Layer 4-7 configurations are provided for testing considerations.

Refer to the Volume 3: LAN (Layer 4-7) Services, page -xxxvii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 17: LAN (Layer 4-7) Service Switch Configurations

Layer 4-7 configurations are provided for testing considerations.

Refer to the Volume 3: LAN (Layer 4-7) Services, page -xxxvii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 18: ACE Configurations

ACE configurations are provided for testing considerations.

Refer to the Volume 3: LAN (Layer 4-7) Services, page -xxxvii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 19: IDSM IPS Configurations

IDSM IPS configurations are provided for testing considerations.

Refer to the Volume 3: LAN (Layer 4-7) Services, page -xxxvii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 20: SAN Configurations

SAN configurations are provided for testing considerations.

Refer to the Volume 4: Storage Area Networking (SAN), page -xxxviii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 21: WAAS ACE Configurations

WAAS configurations are provided for testing considerations.

Refer to the Volume 5: Wide Area Application Services (WAAS), page -xxxviii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 22: WAAS WCCP Configurations

WAAS configurations are provided for testing considerations.

Refer to the Volume 5: Wide Area Application Services (WAAS), page -xxxviii summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 23: GSS Configurations

GSS configurations are provided for testing considerations.

Refer to the Volume 6: Global Site Selector (GSS), page -xxxix summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 24: Bladeswitching Configurations

Blade Server configurations are provided for testing considerations.

Refer to the Volume 7: Bladeswitching, page -xxxix summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 25: Oracle Configurations

Oracle 11i configurations are provided for testing considerations.

Refer to the Volume 8: Applications: Oracle E-Business Suite, page -xxxix summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 26: MS Exchange 2003 Configurations

MS Exchange configurations are provided for testing considerations.

Refer to the Volume 9: Applications: Microsoft Exchange, page -xl summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 27: TIBCO Rendezvous Configurations

TIBCO Rendezvous configurations are provided for testing considerations.

Refer to the Volume 10: Applications: TIBCO Rendezvous, page -xl summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Volume 28: High Availability Configurations

High Availability (Disaster Recovery) configurations are provided for testing considerations.

Refer to the Volume 11: Data Center High Availability, page -xl summary, and associated Cisco Data Center Assurance Program (DCAP) 4.0 document for respective testing information and details.

Results Documents

Table 2 summarizes results documents available in EDCS for Cisco internal audiences. External customers can find externally viewable documents at http://www.cisco.com/go/dcap. External customers may request internal documents from account teams.

Refer to Table 2 for a list of results documents EDCS numbers.

Note There are no output graphs in "Volume 6: Global Site Selector (GSS)" and "Volume 10: Applications: TIBCO Rendezvous" test results so the Customer Facing Plus (CF+) document for these tests is not available.

Table 2 DCAP 4.0 Results Documents

Volume	CF-[1]	CF[2]	CF+[3]	Other[4]
Volume 1: Overview				671343
Volume 2: LAN (L2-3) Infrastructure	671344	671345	671347	Vol. 13-14
Volume 3: LAN (L4-7) Services	671346	671348	671350	Vol. 15-19
Volume 4: Storage Area Networking (SAN)	671351	671352	671357	Vol. 20
Volume 5: Wide Area Application Services (WAAS)	671353	671354	671355	Vol. 21-22
Volume 6: Global Site Selector (GSS)	671356	671358		Vol. 23
Volume 7: Blade Servers	671359	671360	671361	Vol. 24
Volume 8: Applications: Oracle E-Business Suite	671362	671363	671364	Vol. 25
Volume 9: Applications: Microsoft Exchange 2003	671365	671366	671369	Vol. 26
Volume 10: Applications: TIBCO Rendezvous v7.5	671367	671368		Vol. 27
Volume 11: Data Center High Availability	671371	671370	671372	Vol. 28
Volume 12: Appendix				671373
Volume 13: Configurations: LAN Layer 2-3 CSM				667967
Volume 14: Configurations: LAN Layer 2-3 ACE				667968
Volume 15: Configurations: LAN Layer 4-7 CSM				667969
Volume 16: Configurations: LAN Layer 4-7 ACE				667970
Volume 17: Configurations: LAN Layer 4-7 Service Switch				667971
Volume 18: Configurations: ACE				667972
Volume 19: Configurations: IDSM IPS				667973
Volume 20: Configurations: SAN				667974
Volume 21: Configurations: WAAS ACE				667976
Volume 22: Configurations: WAAS WCCP				667977
Volume 23: Configurations: GSS				667978
Volume 24: Configurations: Blade Servers				667979
Volume 25: Configurations: Oracle E-Business Suite				667980
Volume 26: Configurations: Microsoft Exchange 2003				667981
Volume 27: Configurations: TIBCO Rendezvous				667982
Volume 28: Configurations: Data Center High Availability				667988

1. CF- (Customer Facing Minus)—Introductions, expected results, and results.
2. CF (Customer Facing)—Introductions, procedures, expected results, and results.
3. CF+ (Customer Facing Plus)—Introductions, procedures, output graphs, expected results, and results.
4. Other—Available to all audiences, internal and external.

CHAPTER 1

DCAP 4.0 Overview

The Data Center Assurance Program (DCAP) Release 4.0 encompasses testing as outlined in the"Preface". Each volume chapter provides an overview summarizing respective technology coverage and associated topology details. Layer 2-3 and Layer 4-7 technology testing is summariazed in the following sections of this primary overview:

- Layer 2-3 Infrastructure Overview
- Layer 4-7 Services Overview

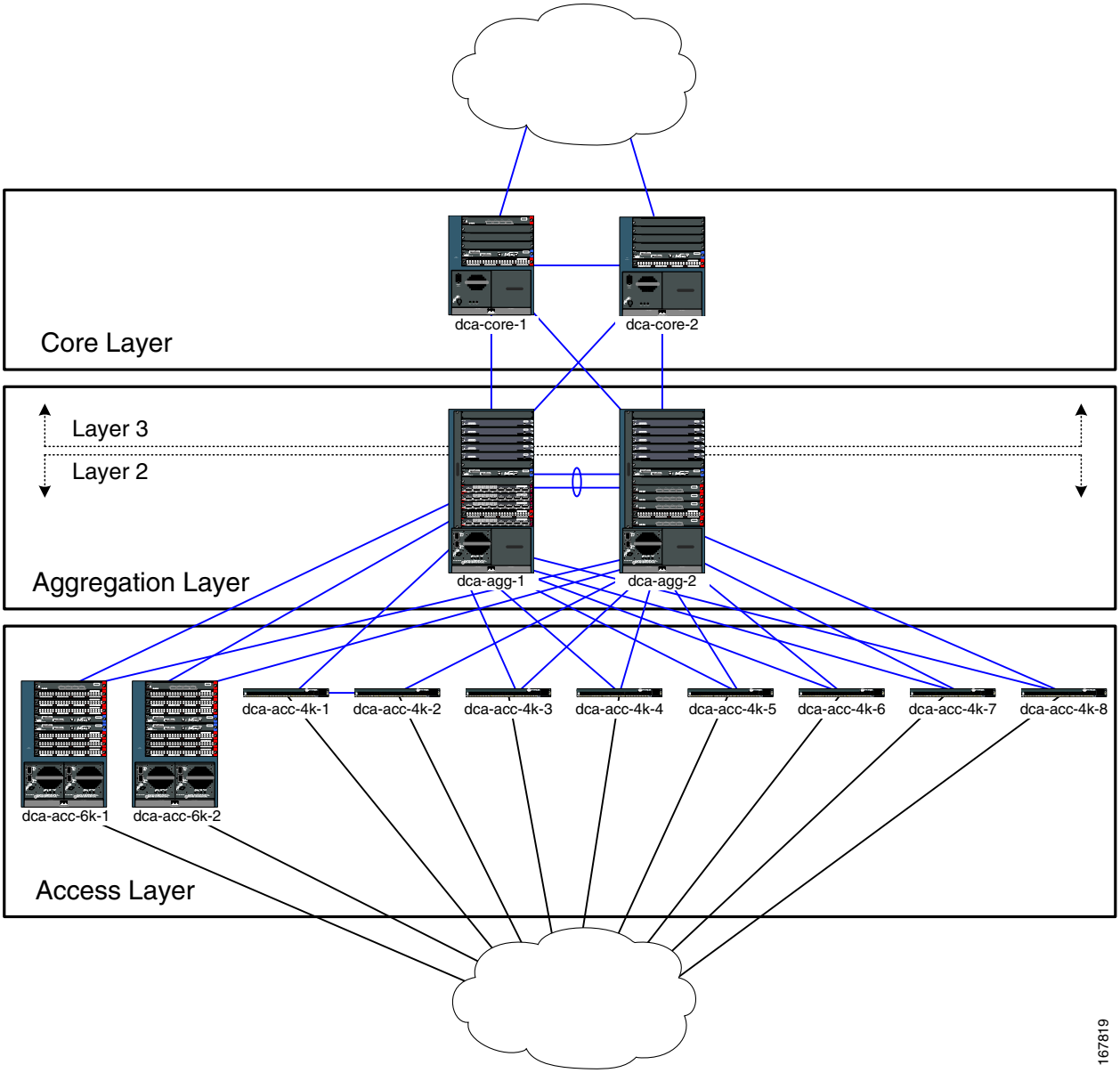
Layer 2-3 Infrastructure Overview

The DCAP 4.0 test topology consists of two separate data centers, DC-A and DC-B. Each data center has its own LAN, SAN and storage components. The tests performed with regards to LAN Layer 2-3 Infrastructure verification were executed against the LAN topology in DC-A. Figure 1-1 shows this portion of the test topology. It is divided into three distinct, logical layers called the Core, Aggregation, and Access Layers offering the Layer 2-3 services listed in Table 1-1.

Table 1-1 Logical Layer Services

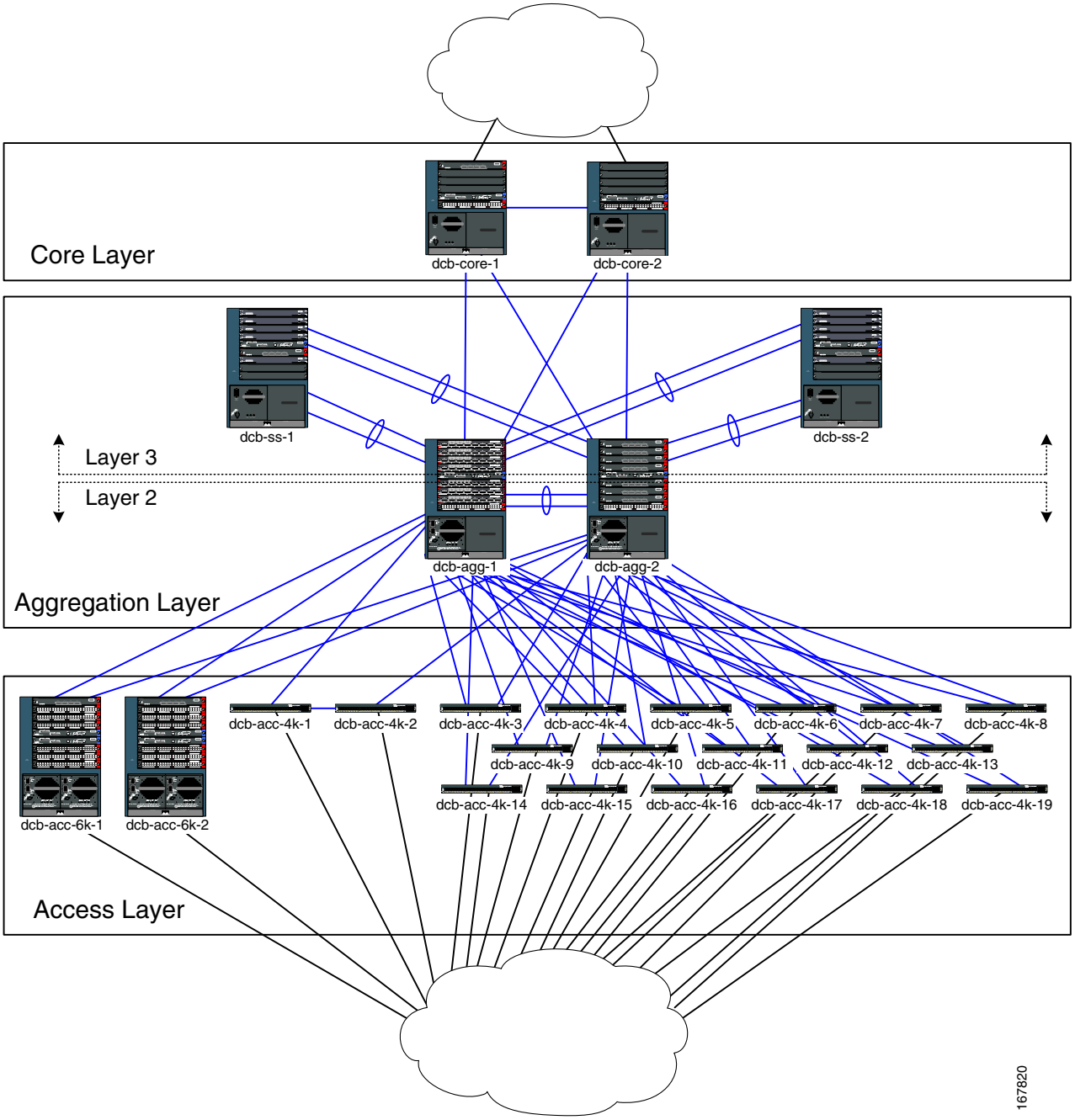
Logical Layer	Services
Core	OSPF, CEF
Aggregation	Default Gateway Redundancy (HSRP), OSPF, Rapid PVST+ Spanning-Tree, UDLD, LACP, 802.1q Trunking
Access	Rapid PVST+ Spanning-Tree, 802.1q Trunking

Figure 1-1 Cisco DCAP 4.0 DCa Topology



The LAN topology in DCB is built a little differently than that in DCA. As will be discussed in a later chapter, the DCB LAN topology is built to accommodate a “Service Chassis” model, in which Layer 4-7 service modules are housed in dedicated switches connected into the Aggregation Layer switches. Like the DCA LAN, the DCB LAN uses both WS-X6704-10GE and WS-X6708-10GE line cards to provide TenGigabitEthernet port density into the Access Layer. The DCB LAN contains two Catalyst 6500 Core Layer switches, two Catalyst 6500 Aggregation Layer switches, two Catalyst 6500 Service Switches, two Catalyst 6500 Access Layer switches, and several Catalyst 4948 Access Layer switches, the bulk of which are present to provide for a scaled spanning-tree environment.

Figure 1-2 Cisco DCAP 4.0 DCb Topology



Layer 2 Topology Overview

Figure 1-1 also shows the demarcation between Layer 2 and Layer 3 in the DCA LAN test topology. There are seven principal devices that operate at Layer 2 in the test topology: dca-agg-1, dca-agg-2, dca-acc-6k-1, dca-acc-6k-2, dca-acc-4k-1, and dca-acc-4k-2. Six additional Catalyst 4948 switches present in the topology provides a more scaled Layer 2 environment, from a spanning-tree perspective.

All interswitch links in the Layer 2 domain are TenGigabitEthernet. For this phase of testing, there are two groups of VLANs. The first group includes VLANs that are actually used for data traffic in the DCAP test plan. There are about 75 VLANs that are actually passing test traffic. In addition to these 75, there are roughly 170 additional VLANs in the DCAP Layer 2 domain that have been included to provide some scaling for Spanning-Tree and HSRP.

Each of the seven devices in the Layer 2 domain participates in Spanning-Tree. The Aggregation Layer device dca-agg-1 is configured as the primary STP root device for all VLANs in the Layer 2 domain, and dca-agg-2 is configured as the secondary STP root. The Spanning-Tree Protocol (STP) that is used in the DCAP topology is PVST+ plus the rapid convergence enhancements of IEEE 802.1w (collectively referred to as Rapid PVST+ or rPVST+).

The Aggregation Layer devices provide a number of services to the data traffic in the network. The Firewall Services Module (FWSM), installed in each of the two Aggregation Layer devices, provides some of these services. In the DCAP topology, the FWSM is operating in multi-context transparent mode and bridges traffic between the outside VLAN to the inside VLAN. As such, only a subset of VLANs (inside VLANs) are propagated down to the Access Layer devices, and the servers that reside on them.

While only a subset of VLANs is carried on the trunks connecting the Access Layer to the Aggregation Layer, the trunk between dca-agg-1 and dca-agg-2 carries all VLANs in the Layer 2 domain. This includes the same subset of inside VLANs that are carried to the Access Layer, their counterpart subset of outside VLANs, as well as a small subset of management VLANs.

Some of these management VLANs carried between dca-agg-1 and dca-agg-2 carry keepalive traffic for the service modules in these two devices. The active and standby pass heartbeat messages between each other so that, should the active become unavailable, the standby can transition itself to take over the active role for those services. If communication between the active and standby peers is lost, and the hardware has not been impacted, an “active/active” condition will likely result. This can wreak havoc on a service-based network and the data traffic that it carries. The reliability of communication between the two peers, then, is important.

The criticality of these heartbeat messages mandates a high level of redundancy for the link carrying these heartbeats. For this reason, two TenGigabitEthernet links are bundled together using LACP to form an etherchannel between dca-agg-1 and dca-agg-2. Having two links provides one level of redundancy. Having these links split between two modules on each device provides an additional level of redundancy.

Layer 3 Topology Overview

Referring again to Figure 1-1, there are four devices that operate at Layer 3 of the OSI stack: dca-core-1, dca-core-2, dca-agg-1, and dca-agg-2.

The Layer 3 portion of the topology is fully meshed with TenGigabitEthernet, with OSPF running as the interior gateway protocol. The devices dca-core-1 and dca-core-2 serve as Area Border Routers (ABR) between Area 0 and Area 10. The link between these two Core Layer devices is in OSPF Area 0. The links between the Core Layer devices and the Aggregation Layer devices are in OSPF Area 10.

In the DCAP test topology, each of the Core Layer devices links up towards the Client cloud. These links are also in Area 0 and this is how the Layer 3 devices in the test topology know about the Client subnets.

The devices dca-agg-1 and dca-agg-2 provide default gateway redundancy via Hot Standby Router Protocol (HSRP). An HSRP default gateway is provided for each of the subnets defined by VLANs in the Layer 2 domain. By configuration, dca-agg-1 is the Active HSRP Router and dca-agg-2 the Standby. Preempt is configured for each VLAN on each of these two devices.

Layer 4-7 Services Overview

There are several Layer 4-7 services that are employed as part of the DCAP 4.0 test topology. They include load balancing, firewalling, SSL offloading, intrusion detection and prevention, global site load balancing and application acceleration.Table 1-2 shows these Layer 4-7 services and the Cisco product line that DCAP uses to provide them.

Table 1-2 Layer 4-7 Solutions Used in DCAP 4.0

Layer 4-7 Service	DCAP Solution
Load balancing	ACE, CSM
Firewall	FWSM
SSL offloading	SSLM, ACE
Global site load balancing	GSS
Application optimization	WAAS
Intrusion prevention/detection	IDSM

Of the Cisco products used in this list, several are service modules:

- ACE – Application Control Engine
- CSM – Content Switching Module
- FWSM – Firewall Services Module
- SSLM – SSL Module
- IDSM – Intrusion Detection Services Module

The testing of these service modules, in a few combinations and two distinct physical topologies, is the focus of this testing volume. There were two basic combinations of these service modules covered in DCAP 4.0 L4-7 testing:

- CSM + FWSM + SSLM + IDSM (+ NAM)
- ACE + FWSM + IDSM (+ NAM)

Note While the Network Analysis Module (NAM) was installed in each of the chassis, it was not subjected to any formal testing as part of DCAP 4.0.

There were two physical topologies tested that are described in more detail below. The CSM-based service module combination was tested in both of these physical topologies, while the ACE-based combination was only tested in the Integrated Switch model.

Integrated Switch Model

The first physical setup involving the service modules places them directly in the same Aggregation Layer chassis that are providing TenGigabitEthernet port density to the Access Layer. This model is deployed in DCa. A pair of Catalyst 6513 switches was used at the Aggregation Layer to house the service modules and the Ethernet linecards. The service modules occupied several of the first 6 slots of each chassis, while the GigabitEthernet and TenGigabitEthernet line cards were placed in slots 9-13.

In this Integrated Switch model, both the CSM-based and ACE-based combinations were tested. Figure 1-3 and Figure 1-4 illustrate these deployments.

Figure 1-3 CSM-Based Integrated Switch Model

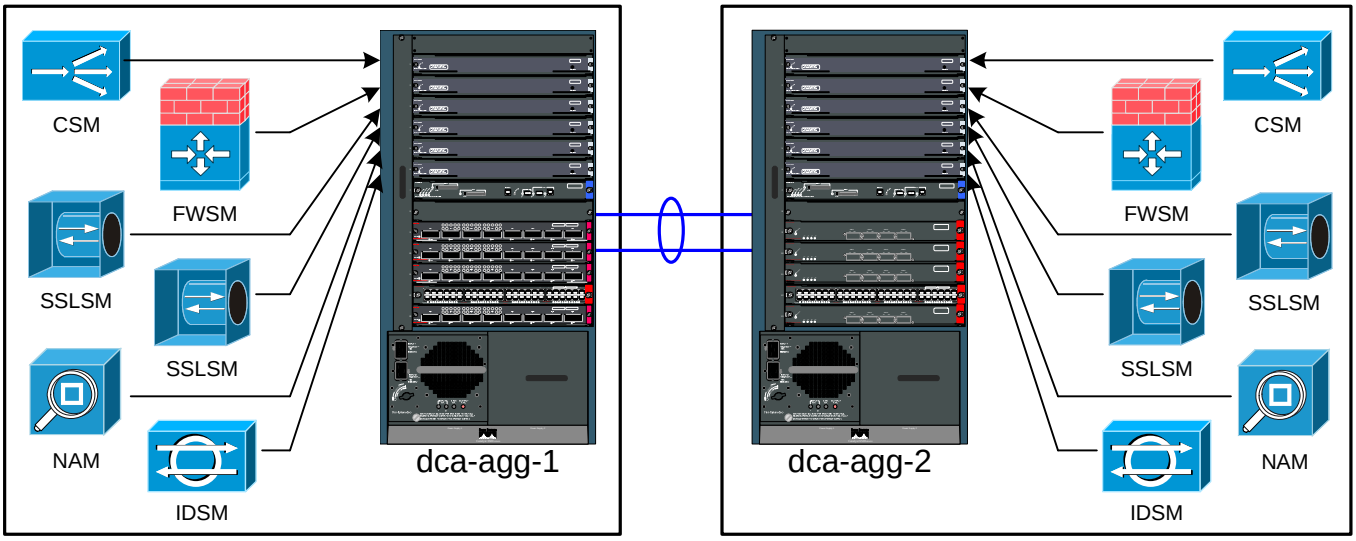
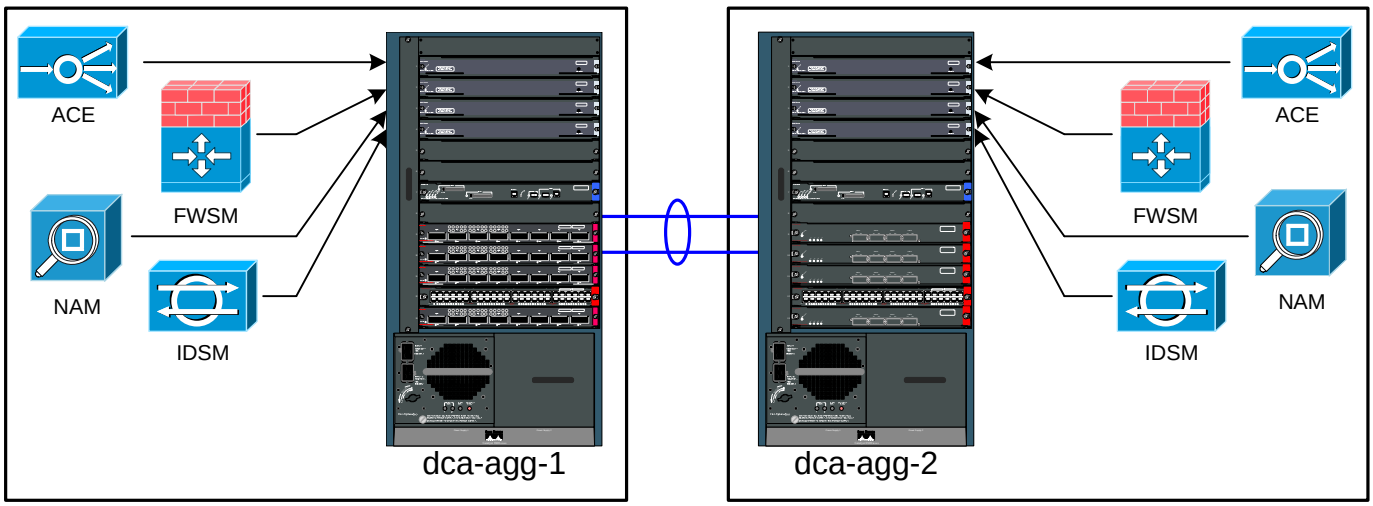


Figure 1-4 ACE-Based Integrated Switch Model



Service Chassis Model

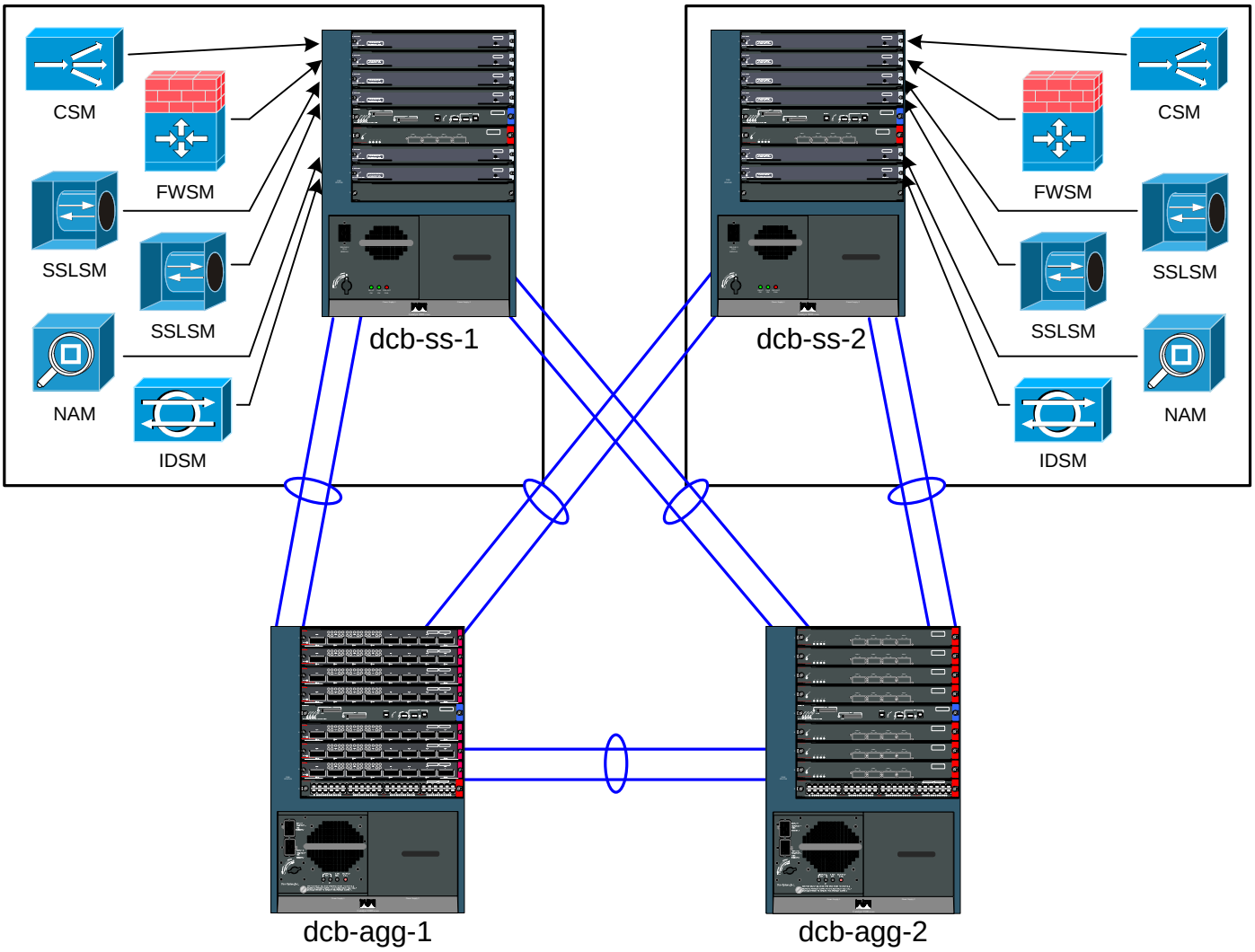
In the DCb LAN, an additional pair of Catalyst 6500 switches, outside of dca-agg-1 and dca-agg-2, was deployed to house the service modules. In the Integrated Switch configuration, the Aggregation Layer devices provide Layer 4-7 services to data center traffic and provide aggregation density to the Access Layer switches. In the Service Chassis configuration, these two functions are cleaved.

These Service Chassis are Catalyst 6509s with one slot used for the Supervisor 720 and a second slot used for a TenGigabitEthernet module (WS-X6704-10GE) to provide connectivity to the Aggregation Layer switches. This leaves a full seven slots available for installing Catalyst 6500 service modules.

With the service modules in their own chassis in the Service Chassis deployment, the Aggregation Layer switch is freed up to provide additional port density to the Access Layer. The Catalyst 6513 still only provides 5 slots of 40 Gbps density, though, since only slots 9-13 are dual-fabric capable, a requirement for the WS-X6704-10GE, WS-X6708-10GE and WS-X748-GE-TX modules. Therefore, a Catalyst 6509 was used as the Aggregation Layer devices, dcb-agg-1 and dcb-agg-2, providing 8 slots for Access Layer port density.

Only the CSM-based bundle was tested in the Service Chassis setup. Figure 1-5 shows the details of this topology in DCb.

Figure 1-5 CSM-Based Service Chassis Model



CSM—Based Integrated Switch Bundle Description

Each of the aggregation devices in the integrated bundle contains an equal number of service modules: one CSM, one FWSM, two SSLMs, one IDSM and a NAM, as illustrated in Figure 1-3. There is a need for communication between the Service Modules not only in the same chassis, but also between chassis, as will be explained below. The 10-Gigabit Etherchannel provides the Layer 2 adjacency needed to facilitate this communication.

There are several modes that the CSM can be configured to operate in, including bridged and routed modes. In this DCAP test setup, it is configured to operate in one-arm mode, which is a variation of routed mode. Essentially, the CSM can be viewed as an appliance connected via a single link to the Aggregation device. Being in routed mode, it serves as a gateway for both client and server traffic.

The CSM is connected, internally, via an etherchannel consisting of four GigabitEthernet interfaces. This etherchannel is a trunk carrying VLANs 301-303. VLAN 301 is used for traffic that needs to be load-balanced without the use of the SSL services provided by the SSLM blades. VLAN 302, which runs on the link connecting the SSLMs to the MSFC, is used for traffic that needs to be load-balanced and also requires the encryption or decryption services of the SSLM blades. VLAN 303 is used to communicate with the peer CSM in the other Aggregation Layer device via the heartbeat messages.

As discussed earlier, there are two CSMs in this test topology, one in dca-agg-1 and one in dca-agg-2. With CSM version 4.2(6), only one CSM can effectively be active at a time. The CSM in dca-agg-1 is configured with a priority such that it is the active CSM, when both CSMs are able to communicate with each other. In steady-state in the DCAP topology, each CSM sends heartbeat messages to its peer every two seconds. If 6 seconds pass between subsequent heartbeat messages, a CSM will consider its peer to be unavailable. If the active CSM stops hearing from the standby, nothing will happen other than learning that the standby is unavailable. If the standby stops hearing from the active, though, it will transition itself to active state and begin to advertise its services and gateways to the network. Because an active/active condition can wreak havoc on a network when both CSMs begin to advertise the same service, the etherchannel between the two Aggregation Layer devices is critical.

There are two modes that the FWSM can operate in, routed and transparent. There is also the option of configuring more than one operational context. Different contexts provide virtual firewalls to different traffic. The FWSMs in the test topology are configured to operate in transparent mode using 31 separate contexts (in addition to the default "system" and "admin" contexts).

In transparent mode, the firewall is actually bridging traffic from an outside VLAN to an inside VLAN, and vice versa. In the DCAP test topology, the outside VLANs used for data are VLANs 1101-1131. The corresponding inside VLANs are 2101-2131. Client traffic whose destination is a real server on VLAN 2101 on the inside, will hit the firewall from the outside on VLAN 1101 and be bridged onto 2101. The opposite will take place in the other direction.

Like the CSM, the FWSM also uses heartbeat messages to communicate with its peer, verifying its existence. The heartbeat messages are sent on VLAN 4 every 2 seconds. If 6 seconds pass between subsequent heartbeat messages, a FWSM will consider its peer to be unavailable. As with the CSM, if the active FWSM stops hearing from the standby, nothing will happen other than it will learn that the standby is unavailable. If the standby stops hearing from the active, though, it will transition itself to active state and begin to advertise its services and gateways to the network. An active/active condition is a dangerous possibility with the FWSM too.

VLAN 5 is used to communicate configuration information between the two peer FWSMs. Outside of certain elements of the "admin" and "system" contexts, there is only one configuration shared between the two FWSMs. The active will use the connection with the standby on VLAN 5 to synchronize the configuration between the two devices (with the active overwriting the standby whenever there are differences). VLAN 5 is also used by the active FWSM to replicate the current connections to the standby. In the event that a failover does occur, the standby will not lose time re-learning all of the connections that the active had established. Note that this is only good for long-term connections.

There are 31 contexts in the FWSM configuration, one for each of the VLANs. They are named "Vlan1101-2101" through "Vlan1131-2131" to reflect the VLANs they are associated with (outside-inside). These contexts govern the traffic that is allowed to access the inside from the outside, and vice versa. In the test topology, each context is essentially the same, save some minor differences. At this point, all contexts are very much promiscuous with regards to what traffic they let through.

VLAN 6 is used for management (telnet) access to the FWSM.

There are four SSLMs in the test topology, two in each of the Aggregation Layer devices. The SSLMs are neither active nor standby; they work in a pool to handle encryption services for data center traffic. Each of the CSMs is configured with a serverfarm called "SSLM" which contains four "real servers." The IP addresses of these real servers are the inband IP addresses of the four SSLMs. When HTTPS traffic comes into the CSM to be load-balanced, it is handed off to this serverfarm and the decryption request is handled by one of the four SSLMs.

Though the four SSLMs are located in separate physical switches (dca-agg-1 and dca-agg-2 or dcb-ss-1 and dcb-ss-2) they are used as if they were in one location. The encryption and decryption requests traverse the interswitch etherchannels, if need be, to reach the necessary SSLM.

ACE—Based Integrated Switch Bundle Configuration

For the tests that were run in the ACE-based model, each of the Aggregation Layer devices in the integrated bundle contains an equal number of Service Modules: one of each ACE, FWSM, IDSM and NAM. Like in the CSM model, there needs to be communication between the service modules not only in the same chassis, but also between chassis, as will be explained below. The 10-Gigabit Etherchannel provides the Layer 2 adjacency needed to facilitate this communication.

There are several modes that the ACE can be configured to operate in, including bridged and routed modes. In the test topology in DCa, the ACE was configured to operate in bridged mode. In bridge mode, the ACE acts as a "bump in the wire" and is not a routed hop. No dynamic routing protocols are required.

When you configure a bridge group on an interface VLAN, the ACE automatically makes it a bridged interface. The ACE supports a maximum of two Layer 2 interface VLANs per bridge group. Because L2 VLANs are not associated with an IP address, they require extended Access Control Lists (ACL) for controlling IP traffic. You can also optionally configure EtherType ACLs for the passing of non-IP traffic. The ACE supports a maximum of 8192 interfaces per system that include VLANs, shared VLANs and BVI interfaces.

The ACE is connected, internally, via an etherchannel consisting of four GigabitEthernet interfaces. This etherchannel is a trunk carrying VLANs 301-303. VLAN 301 is used for traffic that needs to be load-balanced without the use of the SSL services provided by the SSLM blades. VLAN 302, which runs on the link connecting the SSLMs to the MSFC, is used for traffic that needs to be load-balanced and also requires the encryption or decryption services of the SSLM blades. VLAN 303 is used to communicate with the peer CSM in the other Aggregation Layer device via the heartbeat messages.

As discussed earlier, there are two ACE modules in the test topology, one in dca-agg-1 and one in dca-agg-2. Both ACE's are running version A1(6.1), and only one ACE is active at a time. The ACE in dca-agg-1 is configured with a priority such that it is the active ACE, when both ACE modules are able to communicate with each other. In the steady-state in the DCAP topology, each ACE sends heartbeat messages to its peer every 600 milliseconds.

The heartbeat frequency is 600 msec and the heartbeat count is 10. If 6000 msec (6 seconds) seconds pass between subsequent heartbeat messages, an ACE will consider its peer to be unavailable. If the active ACE stops hearing from the standby, nothing will happen other than learning that the standby is unavailable. If the standby stops hearing from the active, though, it will transition itself to active state and begin to advertise its services and gateways to the network. An active/active condition can wreak havoc on a network if both ACE modules begin to advertise the same service. This is why the etherchannel between the two Aggregation Layer devices is so critical.

The ACE sends and receives all redundancy-related traffic (protocol packets, configuration data, heartbeats, and state replication packets) on a dedicated FT VLAN. You can configure a maximum of two ACE modules (peers) in the same Catalyst 6500 switch or in different chassis for redundancy. Each peer module can contain one or more fault-tolerant (FT) groups. Each FT group consists of two members: one active context and one standby context. You cannot use this VLAN for normal traffic. In the DCAP topology in DCa, VLAN 1120 is used as the FT VLAN.

There are two modes that the FWSM can operate in, Routed and Transparent. There is also the option of configuring more than one operational context. Different contexts provide virtual firewalls to different traffic. The FWSMs in the DCAP test topology are configured to operate in transparent mode using 31 separate contexts (in addition to the default "system" and "admin" contexts).

In transparent mode, the firewall is actually bridging traffic from an outside VLAN to an inside VLAN, and vice versa. In the DCAP test topology, the outside VLANs used for data are VLANs 1101-1131. The corresponding inside VLANs are 2101-2131. Client traffic whose destination is a real server on VLAN 2101 on the inside, will hit the firewall from the outside on VLAN 1101 and be bridged onto 2101. The opposite will take place in the other direction.

Like the ACE and CSM, the FWSM also uses heartbeat messages to communicate with its peer, verifying its existence. The heartbeat messages are sent on VLAN 4 every 2 seconds. If 6 seconds pass between subsequent heartbeat messages, a FWSM will consider its peer to be unavailable. As with the ACE and CSM, if the active FWSM stops hearing from the standby, nothing will happen other than it will learn that the standby is unavailable. If the standby stops hearing from the active, though, it will transition itself to active state and begin to advertise its services and gateways to the network. An active/active condition is a dangerous possibility with the FWSM as well.

VLAN 5 is used to communicate configuration information between the two peer FWSMs. Outside of certain elements of the "admin" and "system" contexts, there is only one configuration shared between the two FWSMs. The active will use the connection with the standby on VLAN 5 to synchronize the configuration between the two devices (with the active overwriting the standby whenever there are differences). VLAN 5 is also used by the active FWSM to replicate the current connections to the standby. In the event that a failover does occur, the standby will not lose time re-learning all of the connections that the active had established. Note that this is only good for long-term connections.

There are 31 contexts in the FWSM configuration, one for each of the VLANs. They are named "Vlan1101-2101" through "Vlan1131-2131" to reflect the VLANs they are associated with (outside-inside). These contexts govern the traffic that is allowed to access the inside from the outside, and vice versa. In the DCAP test topology, each context is essentially the same, save some minor differences. At this point, all contexts are very much promiscuous with regards to what traffic they let through.

VLAN 6 is used for management (telnet) access to the FWSM.

Service Switch Configuration

Logically, the service module bundle in the service chassis deployment handles traffic in the same manner as in the integrated bundle. The service module configurations are nearly identical, down to the VLAN. Physically, the service chassis is quite different, as can be seen in Figure 3-3, above.

There are two service chassis, dcb-ss-1 and dcb-ss-2. As in the integrated switches, dca-agg-1 and dca-agg-2, these service chassis have a single CSM, a single FWSM, two SSLMs, and IDSM and a NAM. Internally, they are connected in the same way. Externally, each service chassis is dual-homed to the pair of Aggregation Layer devices, dcb-agg-1 and dcb-agg-2. If they were connected to the Aggregation Layer via a single etherchannel, and that channel was broken, a failover event would occur with regards to the active CSM and FWSM. It would take 6 seconds plus the amount of time necessary to re-build any TCP connections to restore traffic if such a failover event occurred. With dual-homing, the sub-second convergence time of the Rapid PVST+ Spanning-Tree protocol can be relied on to ensure that such a failover event does not occur.

The four etherchannels connecting the service chassis to the Aggregation Layer switches carry all of the inside and outside data VLANs, 1101-1131 and 2101-2131, as well as the VLANs necessary for connection replication, redundancy, and out-of-band management. VLAN 10 is also configured in order to facilitate OSPF adjacency between the two service chassis and the two Aggregation Layer devices and help make the networks residing on dcb-ss-1 and dcb-ss-2 known to the rest of the DCa LAN topology. The Spanning-Tree configuration remains unchanged, with dcb-agg-1 as the primary STP root and dcb-agg-2 as the secondary root.

It is important to mention some changes that are necessary at Layer 3 in order to support the service chassis model. For those inside VLANs whose networks are advertised out to the world, the service chassis share default gateway duties via HSRP. The device dcb-ss-1 is configured with the higher HSRP priority and would thus be the active HSRP router in a steady-state condition, with dcb-ss-2 waiting as standby. (For those VLANs carrying traffic that is not serviced by the service chassis bundle, dcb-agg-1 and dcb-agg-2 share the HSRP responsibilities, as in the integrated bundle setup.)

Layer 4-7 Services Test Results

There were 44 tests that are covered in this results volume. They are broken up into several different sections. Some of the testing was executed on individual service modules while some was done against the combination of service modules. The following sections provide the results to DCAP 4.0 Layer 4-7 security testing.

CSM-Based Integrated Switch Bundle

Tests in the "Layer 4-7 CSM" section on page 2-1 focus on traffic flows that touch multiple service modules. The service module combination that was used for this section was CSM+FWSM+SSLM+IDSM+NAM. Figure 1-3 illustrates the relevant topology for this section.

ACE-Based Integrated Switch Bundle

Tests in the "Layer 4-7 ACE" section on page 3-1 focus on the traffic flows that touch multiple service modules. The service module combination that was used for this section was ACE+FWSM+IDSM+NAM. Only the subset of the CSM-based bundle tests that include the ACE were run in this section. Figure 1-4 illustrates the relevant topology for this section.

CSM-Based Service Chassis Bundle

Tests in the "Layer 4-7 Services Switch" section on page 4-1 focus on the traffic flows that touch multiple service modules. The service module combination that was used for this section was CSM+FWSM+SSLM+IDSM+NAM in the service chassis model. Figure 1-5 illustrates the relevant topology for this section.

Application Control Engine (ACE)

Tests in the "ACE" section on page 5-1focus on the functionality of the ACE service module, operating in the DCAP environment. The results for additional tests run by the Safe Harbor team against the software version tested in DCAP 4.0 are available in the DCAP 4.0 Appendix.

Intrusion Detection Services Module (IDSM)

Tests in the "IDSM IPS" section on page 6-1 focus on the functionality of the IDSM service module, operating in the DCAP environment. The results for additional tests run by the Safe Harbor team against the software version tested in DCAP 4.0 are available in the DCAP 4.0 Appendix.

CHAPTER 2

Layer 2-3 Infrastructure with CSM

Layer 2-3 CSM testing reflects a design in which the Content Switching Module (CSM) and the SSL Service Modules (SSLSM) were used in the Aggregation Layer switches. In Layer 2-3 Infrastructure with ACE, page 3-1 testing, the CSM/SSLSM combination was replaced with the Application Control Engine (ACE). Layer 2-3 Infrastructure with ACE testing is a subset of testing reported in this chapter.

Test Results Summary

Table 2-1 on page 2-1 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 2-1 on page 2-1 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 2-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Baseline, page 2-5	Baseline, page 2-5	1. Topology Baseline (CSM Setup)	
Baseline, page 2-5	CLI Functionality, page 2-7	1. CLI Parser Functionality Using SSHv1 on 4948 (CSM Setup) 2. CLI Parser Functionality Using SSHv1 (CSM Setup) 3. CLI Parser Functionality Using SSHv2 on 4948 4. CLI Parser Functionality Using SSHv2 (CSM Setup)	CSCsc81109
Baseline, page 2-5	Device Access, page 2-10	1. Repeated Logins Using SSH Version 1 (CSM Setup) 2. Repeated Logins Using SSH Version 2 (CSM Setup)	

Table 2-1 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Baseline, page 2-5	Device Management, page 2-11	1. General On-Line Diagnostics (GOLD) 2. Local SPAN (CSM Setup) 3. Remote SPAN (rSPAN) (CSM Setup) 4. SNMP MIB Tree Walk 5. Upgrade Firewall Services Module (FWSM) 6. Upgrade of Supervisor 720 System in Access Layer 7. Upgrade of Supervisor 720 System in Aggregation Layer (CSM Setup) 8. Upgrade of Supervisor 720 System in Core Layer (CSM Setup) 9. Upgrade of Catalyst 4948-10GE System in Access Layer (CSM Setup)	
Baseline, page 2-5	Security, page 2-21	1. Malformed SNMP Polling 2. Malformed SSH Packets (CSM Setup) 3. NMAP Open Port Scan	
Baseline, page 2-5	Traffic Forwarding, page 2-24	1. Scaled FIB Consistency (CSM Setup) 2. Zero Packet Loss (CSM Setup)	
Layer 2 Protocols, page 2-27	Link Aggregation Control Protocol (LACP), page 2-27	1. LACP Basic Functionality 2. LACP Load Balancing (CSM Setup)	
Layer 2 Protocols, page 2-27	Spanning-Tree Protocol (STP), page 2-29	1. Rapid PVST+ Basic Functionality (CSM Setup) 2. Root Guard	
Layer 2 Protocols, page 2-27	Trunking, page 2-34	1. 802.1q Trunking Basic Functionality	
Layer 2 Protocols, page 2-27	Unidirectional Link Detection (UDLD), page 2-35	1. UDLD Detection on 10-Gigabit Ethernet Links	
Layer 3 Protocols, page 2-37	Hot Standby Router Protocol (HSRP), page 2-37	1. HSRP Basic Functionality	
Layer 3 Protocols, page 2-37	Open Shortest Path First (OSPF), page 2-38	1. OSPF Database Verification (CSM Setup) 2. OSPF Pagent Convergence Test Second Aggregate Layer 3. OSPF Pagent Router Flap Test Second Aggregate Layer 4. OSPF Pagent Verify Test Second Aggregate Layer 5. OSPF Route Summarization (CSM Setup)	

Table 2-1 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Negative, page 2-44	Hardware Failure, page 2-45	1. Access Layer Supervisor Failover Using SSO with NSF (CSM Setup) 2. Failure of Etherchannel Module on dca-agg-1 (CSM Setup) 3. Failure of Etherchannel Module on dca-agg-2 (CSM Setup) 4. HSRP Failover with Fast Timers 5. HSRP Recovery From System Failure 6. Repeated Reset of Standby Supervisor in Access Layer (CSM Setup) 7. Reset of Aggregation Layer Device dca-agg-1 (CSM Setup) 8. Reset of Aggregation Layer Device dca-agg-2 (CSM Setup) 9. Reset of Core Layer Device dca-core-1 (CSM Setup) 10. Reset of Core Layer Device dca-core-2 (CSM Setup) 11. Spanning-Tree Primary Root Failure and Recovery (CSM Setup)	CSCsk60108

Table 2-1 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Negative, page 2-44	Link Failure, page 2-62	1. Failure of 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2 (CSM Setup) 2. Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1 (CSM Setup) 3. Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1 (CSM Setup) 4. Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2 (CSM Setup) 5. Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2 (CSM Setup) 6. Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1 (CSM Setup) 7. Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2 (CSM Setup) 8. Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1 (CSM Setup) 9. Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2 (CSM Setup) 10. Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2 (CSM Setup) 11. Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1 (CSM Setup) 12. Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2 (CSM Setup) 13. Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2 (CSM Setup) 14. Network Resiliency Test (CSM Setup)	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Baseline

The baseline tests are focused on various aspects of administering the devices in the DCAP test topology, as well as the verification of the most basic features such as distributed forwarding and security. Baseline tests verify network is in working order prior to starting testing and quantify steady state network performance.

This section contains the following topics:

Baseline

In all of DCAP testing, system resources of all the Layer 2/3 devices in the test topology are monitored, including CPU and memory utilization. When an issue is suspected, manifest as a sustained CPU spike or consumed memory for example, it is helpful to have a steady-state baseline of what the network resources look like for comparison purposes. The tests in this section help to establish a baseline level of expected behavior so that real problems can be more easily identified.

This section contains the following topics:

Topology Baseline (CSM Setup)

This test verified that the network is in an operational state. An initial snapshot of the current network state is taken. Background traffic is left running for approximately two hours. At the end of this time the current network state is compared to the baseline snapshot taken at the beginning of this test. This comparison verifies the network is ready for testing by examining features, protocols and interface counters for discrepancies.

It also provides a baseline of what the system resources (CPU and memory) look like while the traffic that is used in the tests is running. This is useful for comparison purposes during the other tests.

Test Procedure

The procedure used to perform the Topology Baseline (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Baseline all CDP neighbor relationships. Run the CDP crawler script verifying all expected CDP neighbors are reported.

The purposed of the CDP crawler script is to crawl through the network continuously, noting any changes that occur between traversals in CDP information. It parses information gathered from select CDP and IOS commands.

Step 3 Baseline all EtherChannel members. Run the channel crawler script verifying that all interfaces expected to be in channels are reported.

The purpose of the channel crawler script is to run through a network and verify that EtherChannels are in a proper state. It parses information gathered from select EtherChannel and IOS commands.

Step 4 Baseline all trunk interfaces. Run the trunk crawler script verifying that all expected trunking interfaces, configuration, and status are reported.

The purpose of the trunk crawler script is to run through a network and verify that trunking is in a proper state. It parses information gathered from select trunking and IOS commands.

Step 5 Baseline all interface states and counters. Run the interface crawler script recording interface counters and states.

The interface crawler script crawls through a network continually. All up/up interfaces are checked for various errors. Initially all non zero error counters will be logged, then any counters that increment from that point on.

Step 6 Baseline all interface UDLD states. Run the UDLD crawler script recording the UDLD state of all interfaces.

The UDLD crawler script gathers a list of UDLD ports from a list of devices and traverses their neighbors continuously, checking for UDLD problems or inconsistencies. It parses information gathered from select UDLD and IOS commands.

Step 7 Baseline all linecards used in the topology. Run the module crawler script recording module counters and state.

The module crawler script gathers a list of modules from a list of devices and looks for problems or inconsistencies. It parses information gathered from select module and IOS commands.

Step 8 Begin the test traffic. Allow it to run for two hours.

Step 9 Execute the CDP crawler script to verify that the CDP feature is operating in the Data Center test network as it was before background traffic was started.

Step 10 Execute the channel crawler script to verify that the EtherChannel feature is operating in the Data Center test network as it was before background traffic was started.

Step 11 Execute the trunk crawler script to verify that the trunking feature is operating in the Data Center test network as it was before background traffic was started.

Step 12 Execute the interface crawler script to verify that the basic functionality of the interface is operating in the Data Center test network as it was before background traffic was started.

Step 13 Execute the UDLD crawler script to verify that the UDLD feature is operating in the Data Center test network as it was before background traffic was started.

Step 14 Execute the module crawler script to verify that the line cards in the Data Center test network are still operating correctly after background traffic was started.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that there will be no change in the test topology during the baseline period.
- We expect no CPU or memory problems.

Results

Topology Baseline (CSM Setup) passed.

CLI Functionality

Command Line testing robustly exercises the command line interface (CLI) of a router. The testing walks the parser tree, executing completed commands and filling in options as it comes to them. Certain branches of the parser tree were left out due to time constraints of the testing (eg. show tag-switching tdp, show mpls).

This section contains the following topics:

CLI Parser Functionality Using SSHv1 on 4948 (CSM Setup)

An automated script was used to test the valid **show** and **clear** commands on dca-acc-4k-1. SSH version 1 was used as the access protocol.

Test Procedure

The procedure used to perform the CLI Parser Functionality Using SSHv1 on 4948 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin executing the **show** and **clear** commands on the device under test.

Step 3 Stop background scripts to collect final status of network devices and analyze for error.

Step 4 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all commands to be executed without negative impact on the DUT.
- We expect no CPU or memory problems.

Results

CLI Parser Functionality Using SSHv1 on 4948 (CSM Setup) passed.

CLI Parser Functionality Using SSHv1 (CSM Setup)

An automated script was used to test the valid **show** and **clear** commands on dca-agg-2. The commands that were tested were a select subset of those tested in the full Native IOS Safe Harbor releases. These commands were chosen based on their relation to differentiating hardware and software features between the traditional Safe Harbor Native IOS topologies and the DCAP topology. SSH version 1 was used as the access protocol.

Test Procedure

The procedure used to perform the CLI Parser Functionality Using SSHv1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin executing the **show** and **clear** commands on the device under test.

Step 3 Stop background scripts to collect final status of network devices and analyze for error.

Step 4 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all commands to be executed without negative impact on the DUT.
- We expect no CPU or memory problems.

Results

CLI Parser Functionality Using SSHv1 (CSM Setup) passed.

CLI Parser Functionality Using SSHv2 on 4948

An automated script was used to test the valid **show** and **clear** commands on dca-acc-4k-2. SSH version 2 was used as the access protocol.

Test Procedure

The procedure used to perform the CLI Parser Functionality Using SSHv2 on 4948 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin executing the **show** and **clear** commands on the device under test.

Step 3 Stop background scripts to collect final status of network devices and analyze for error.

Step 4 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all commands to be executed without negative impact on the DUT.
- We expect no CPU or memory problems.

Results

CLI Parser Functionality Using SSHv2 on 4948 passed.

CLI Parser Functionality Using SSHv2 (CSM Setup)

An automated script was used to test the valid **show** and **clear** commands on dca-agg-2. The commands that were tested were a select subset of those tested in the full Native IOS Safe Harbor releases. These commands were chosen based on their relation to differentiating hardware and software features between the traditional Safe Harbor Native IOS topologies and the DCAP topology. SSH version 2 was used as the access protocol.

Test Procedure

The procedure used to perform the CLI Parser Functionality Using SSHv2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin executing the **show** and **clear** commands on the device under test.

Step 3 Stop background scripts to collect final status of network devices and analyze for error.

Step 4 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all commands to be executed without negative impact on the DUT.
- We expect no CPU or memory problems.

Results

CLI Parser Functionality Using SSHv2 (CSM Setup) passed.

Device Access

The DCAP test topology includes dedicated out-of-band management links on all of the network devices. The access protocol used on all of these devices is SSH, for security purposes. These tests stress the access protocols used.This section contains the following topics:

- Repeated Logins Using SSH Version 1 (CSM Setup), page 2-10
- Repeated Logins Using SSH Version 2 (CSM Setup), page 2-11

Repeated Logins Using SSH Version 1 (CSM Setup)

The device dca-agg-2 was subjected to 1000 login attempts, using version 1 of the SSH protocol, from each of six iterations of the login script. This was done to max out the available VTY interfaces on dca-agg-2. The full profile of background traffic (HTTP and FTP requests) was running during this test.

Test Procedure

The procedure used to perform the Repeated Logins Using SSH Version 1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the HTTP and FTP background traffic is running.

Step 3 Verify that dca-agg-2 is configured for ssh login using the **show ip ssh** command.

The **show ip ssh** command should show **SSH Enabled—version 1.99** in the output.

Step 4 Initiate 6 iterations of the test script. Each iteration will attempt to log into dca-agg-2 1000 times, successively, using SSH version 1.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that there will be no system error messages resulting from the multiple, repeated SSH login attempts.
- We expect no CPU or memory problems.

Results

Repeated Logins Using SSH Version 1 (CSM Setup) passed.

Repeated Logins Using SSH Version 2 (CSM Setup)

The device dca-agg-1 was subjected to 1000 login attempts, using version 2 of the SSH protocol, from each of six iterations of the login script. This was done to max out the available VTY interfaces on dca-agg-1. The full profile of background traffic (HTTP and FTP requests) was running during this test.

Test Procedure

The procedure used to perform the Repeated Logins Using SSH Version 2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the HTTP and FTP background traffic is running.

Step 3 Verify that dca-agg-1 is configured for ssh login using the **show ip ssh** command.

The **show ip ssh** command should show **SSH Enabled—version 1.99** in the output.

Step 4 Initiate 6 iterations of the test script. Each iteration will attempt to log into dca-agg-1 1000 times, successively, using SSH version 2.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that there will be no system error messages resulting from the multiple, repeated SSH login attempts.
- We expect no CPU or memory problems.

Results

Repeated Logins Using SSH Version 2 (CSM Setup) passed.

Device Management

Simple Network Management Protocol (SNMP) is an application-layer protocol designed to facilitate the exchange of management information between network devices. The tests in this section cover some of the common procedures and features used in the normal operation of a network, including the upgrading of network devices and the use of various features that may be used in troubleshooting.

This section contains the following topics:

General On-Line Diagnostics (GOLD)

General online diagnostics (GOLD) is a software tool that tests and verifies the hardware functionality of the supervisor engine, modules, and switch while the switch is connected to a live network. There are disruptive and non disruptive online diagnostic tests, including a subset of the GOLD tests that are run upon bootup of a hardware component. These are referred to as bootup diagnostics and are run during bootup, module OIR, or switchup to a redundant supervisor.

Each device in the data center topology is configured for a complete diagnostics run on bootup. This test verifies that each device in the data center topology is configured to run complete diagnostics on bootup, and that the complete set of diagnostics was run on each module at the last boot event.

Test Procedure

The procedure used to perform the General On-Line Diagnostics (GOLD) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Log into dca-core-1 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 3 On dca-core-1, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 4 Log into dca-core-2 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 5 On dca-core-2, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 6 Log into dca-agg-1 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 7 On dca-agg-1, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 8 Log into dca-agg-2 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 9 On dca-agg-2, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 10 Log into dca-acc-6k-1 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 11 On dca-acc-6k-1, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 12 Log into dca-acc-6k-2 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 13 On dca-acc-6k-2, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the complete set of online diagnostics to have run on all modules in the systems under test, as configured.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

General On-Line Diagnostics (GOLD) passed.

Local SPAN (CSM Setup)

Local SPAN selects network traffic to send to a network analyzer. SPAN should not affect the switching of network traffic on source ports or VLAN's. SPAN sends a copy of the packets received or transmitted by the source ports and VLAN's to a destination port dedicated for SPAN use.

This test verified that normal traffic forwarding was maintained when a local SPAN session was configured on dca-acc-6k-2. Interface TenGigabit Ethernet 1/1 was used as the SPAN source. The destination was a locally installed Network Analysis Module (NAM). The network was monitored for traffic irregularities and the DUT was monitored for CPU or memory stability.

Test Procedure

The procedure used to perform the Local SPAN (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On dca-agg-1, use the **show monitor** command to verify that there are no SPAN sessions present.

Step 3 Configure the SPAN source to be interface Te1/1 using the**monitor session 1 source interface Te1/1 both** command. By specifying both, the session will SPAN ingress and egress traffic on Te1/1.

Step 4 Configure the SPAN destination to be interface Gi2/41 using the **monitor session 1 destination interface Gi2/41** command.

Step 5 Clear the traffic counters on dca-acc-6k-2 and dca-agg-1 using the **clear counters** command.

Step 6 Begin the capture session on the Knoppix server.

Step 7 Run the background test traffic for a period of 10 minutes.

Step 8 When the background test traffic finishes, verify that it does not report any more than the normal amount of errors.

The script used to run the background test traffic will report statistics in the form of HTTP return codes. The Zero Packet Loss test indicates that the normal number of errors is below 0.01% (comparing, in that test, 500 return codes to 200 return codes).

Step 9 Compare the counters of the SPAN source interface with those of the SPAN destination interface using the **show interface***interface***counters** command.

The SPAN source is monitoring both transmit and receive of the source interface. The SPAN destination interface egress counters should reflect the combination of both directions of traffic on the SPAN source.

Step 10 Look for any errors on the SPAN destination interface using the **show interfaces Gi2/41** command.

Step 11 Remove the SPAN configuration from dca-agg-1 using the **no monitor session 1** command.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the SPAN utility will operate soundly under load.
- We expect that the SPAN utility will not interfere with normal network traffic.
- We expect no CPU or memory problems.

Results

Local SPAN (CSM Setup) passed.

Remote SPAN (rSPAN) (CSM Setup)

With remote SPAN, the SPAN destination is a VLAN, rather than a physical interface. This VLAN is configured as a remote VLAN throughout the network. Traffic that is copied to the SPAN VLAN is tagged with that VLAN ID and sent through the network to a traffic analyzer attached to a network device that is remote to the SPAN source.

This test verified that normal traffic forwarding was maintained when a remote SPAN session was configured on dca-agg-1. Interface Te9/4 was used as the SPAN source. The destination was remote-vlan 900. This VLAN is configured throughout the Layer 2 domain in the DCAP test network. The traffic collector was a locally installed Network Analysis Module (NAM). This server was running the tethereal program to capture the traffic. The network was monitored for traffic irregularities and the DUT was monitored for CPU or memory stability.

Test Procedure

The procedure used to perform the Remote SPAN (rSPAN) (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that VLAN 900 is configured on all the DCAP devices in the Layer 2 domain, and that it is a remote VLAN using the **show vlan id 900** command.

VLAN 900 should be present on dca-agg-1, dca-agg-2, dca-acc-6k-1, dca-acc-6k-2, dca-acc-4k-1, and dca-acc-4k-2. In the output for each of these devices, the Remote SPAN VLAN field should indicate enabled.

Step 3 On dca-agg-1, use the **show monitor** command to verify that no SPAN sessions are present.

There may be a monitor session with ID=1 present on either of the Aggregation layer devices in the DCAP test topology as a result of having service modules in the system. If that is the case, session ID=2 may be used.

Step 4 On dca-agg-1, configure the SPAN source to be interface Te9/5 using the**monitor session 2 source interface Te9/5 both** command. By specifying **both**, the session will SPAN ingress and egress traffic on Te9/5.

Step 5 Configure the SPAN destination to be remote SPAN VLAN 900 using the **monitor session 2 destination remote vlan 900** command.

Step 6 On device dca-acc-6k-2, verify that no SPAN sessions are present using the **show monitor** command.

Step 7 On device dca-acc-6k-2, configure the SPAN source to be remote SPAN VLAN 900 using the**monitor session 1 source remote vlan 900** command.

Step 8 Configure the SPAN destination to be interface Gi2/41 using the **monitor session 1 destination interface Gi2/41** command.

Step 9 Clear the traffic counters on dca-agg-1 and dca-acc-6k-2 using the **clear counters** command.

Step 10 Begin the capture session on the Knoppix server.

Step 11 Run the background test traffic for a period of 10 minutes.

Step 12 When the background test traffic finishes, verify that it does not report any more than the normal amount of errors.

The script that is used to run the background test traffic will report statistics in the form of HTTP return codes. The Zero Packet Loss test indicates that the normal number of errors is below one percent (comparing, in that test, 500/400/402 return codes to 200 return codes).

Step 13 Compare the counters of the SPAN source interface (Te9/4 on dca-agg-1) with those of the SPAN destination interface (Gi2/41 on dca-acc-6k-2) using the **show interface***interface***counters** command.

The SPAN source is monitoring both transmit and receive of the source interface. The SPAN destination interface egress counters should reflect the combination of both directions of traffic on the SPAN source.

It is important to note that the SPAN source interface is a TenGigabit Ethernet interface and that the destination interface is only GigabitEthernet. Packet loss is expected.

Step 14 Look for any errors on the SPAN destination interface using the **show interfaces Gi2/41** command on dca-acc-6k-2.

It is important to note that the SPAN source interface is a TenGigabit Ethernet interface and that the destination interface is only GigabitEthernet. Packet loss is expected.

Step 15 Remove the SPAN configurations from dca-agg-1 and dca-acc-6k-2 using the **no monitor session***session_id* command.

Step 16 Stop background scripts to collect final status of network devices and analyze for error.

Step 17 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the SPAN utility will operate soundly under load.
- We expect that the SPAN utility will not interfere with normal network traffic.
- We expect no CPU or memory problems.

Results

Remote SPAN (rSPAN) (CSM Setup) passed.

SNMP MIB Tree Walk

Simple Network Management Protocol (SNMP) is ubiquitous as a tool used by network administrators to manage network performance, find and solve network problems, and plan for network growth.

This test verified that an SNMP walk of the MIB tree of dca-agg-1 did not cause any memory loss, tracebacks, or reloads. From a server, five version 1 SNMP walks were performed.

Test Procedure

The procedure used to perform the SNMP MIB Tree Walk test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 If the background test traffic is not already running, start it now.

Step 3 Verify the SNMP configuration of dca-agg-1 using the **show running-config** command.

Step 4 From the server CLI perform five SNMP walks on the DUT using the **snmpwalk** utility.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect no tracebacks or crashes to occur on the DUT.
- We also expect no memory loss to occur.

Results

SNMP MIB Tree Walk passed.

Upgrade Firewall Services Module (FWSM)

This test verified the ability for code to be upgraded for the latest version of Safe Harbor certified FWSM code to the version of FWSM code that is under test.

In phase 4 of DCAP the version under test is 3.1(4)0 and the upgrade was performed from version 2.3(3.2)

Test Procedure

The procedure used to perform the Upgrade Firewall Services Module (FWSM) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Record the start time of this test using the **show clock** command on dca-agg-1.

Step 3 Verify that the FWSM in dca-agg-1 is running the old FWSM image using the **show module 1** command.

In this run, the FWSM is running the FWSM image currently under test.

Step 4 Verify that the FWSM image under test is on the proper file device on dca-agg-1 using the **dir sup-bootflash:** command.

Step 5 Use the **show running-config | include tftp-server** command to verify that dca-agg-1 is configured to be a TFTP server for that image. This will make the image downloadable to the FWSM directly from the Supervisor.

Step 6 Set up a session between the supervisor engine and the FWSM using the **session slot 1 processor 1** command.

Step 7 Verify connectivity from the FWSM to the supervisor using the **ping** command to the loopback address of the supervisor, 127.0.0.71.

Step 8 Use the **copy tftp://127.0.0.71/***image_name***flash:** command to download the new image from the TFTP server on the supervisor. Or alternatively copy the image from an external tftpserver

Step 9 Issue the **reload** command on the FWSM to reboot the blade.

Step 10 Once the FWSM has come back online, verify that the new image is running using the **show module 1** command.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the upgrade process on the FWSM platform will proceed smoothly and without error.

Results

Upgrade Firewall Services Module (FWSM) passed.

Upgrade of Supervisor 720 System in Access Layer

This test verified the ability for code to be upgraded for the latest version of Safe Harbor certified SXF code to the version of SXF that is under test. Access layer device dca-acc-6k-1 was upgraded from 12.2(18)SXF7 Native IOS to 12.2(18)SXF9 Native IOS to ensure that all hardware and configurations at the access layer were upgraded without issue.

Test Procedure

The procedure used to perform the Upgrade of Supervisor 720 System in Access Layer test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Record the start time of this test using the **show clock** command.

Step 3 Verify that dca-acc-6k-1 is running the old Native Cisco IOS image using the **show version** command.

Step 4 Verify that the Supervisor 720 image under test is on the proper file devices on dca-acc-6k-1 using the **dir disk0:** and **dir slavedisk0:** commands.

The device dca-acc-6k-1 is configured with dual supervisors. It is therefore necessary that each of these supervisors has the new image in their respective filesystems.

Step 5 Use the **show running-config | include boot** command to verify that the boot string points to the proper device and filename for the test image. If any changes are necessary, make them and then save them to NVRAM when done.

Step 6 Issue the **reload** command on dca-acc-6k-1, causing both supervisors to reboot. Report any error messages seen during reload.

Step 7 Use the **show module** and **show version** commands to verify that dca-acc-6k-1 came online successfully and that the new image is running.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the upgrade process from SXF to SXF on the Supervisor 720 platform will proceed smoothly and without error.

Results

Upgrade of Supervisor 720 System in Access Layer passed.

Upgrade of Supervisor 720 System in Aggregation Layer (CSM Setup)

This test verified the ability for code to be upgraded for the latest version of Safe Harbor certified SXF code to the version of SXF that is under test. Aggregation layer device dca-agg-1 was upgraded from 12.2(18)SXF7 Native IOS to 12.2(18)SXF9 Native IOS to ensure that all hardware and configurations at the core layer were upgraded without issue.

Test Procedure

The procedure used to perform the Upgrade of Supervisor 720 System in Aggregation Layer (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Record the start time of this test using the **show clock** command.

Step 3 Verify that the dca-agg-1 is running the old Native Cisco IOS image using the **show version** command.

Step 4 Verify that the Supervisor 720 image under test is on the proper file device on dca-agg-1 using the **dir disk0:** command.

Step 5 Use the **show running-config | include boot** command to verify that the boot string points to the proper device and filename for the test image. If any changes are necessary, make them and then save them to NVRAM when done.

Step 6 Issue the **reload** command on dca-agg-1, causing the supervisor to reboot. Report any error messages seen during reload.

Step 7 Use the **show module** and **show version** commands to verify that dca-agg-1 came online successfully and that the new image is running.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the upgrade process from SXF to SXF on the Supervisor 720 platform will proceed smoothly and without error.

Results

Upgrade of Supervisor 720 System in Aggregation Layer (CSM Setup) passed with exception. The following exceptions were noted: ">.

Upgrade of Supervisor 720 System in Core Layer (CSM Setup)

This test verified the ability for code to be upgraded for the latest version of Safe Harbor certified SXF code to the version of SXF that is under test. Core layer device dca-core-1 was upgraded from 12.2(18)SXF7 Native IOS to 12.2(18)SXF9 Native IOS to ensure that all hardware and configurations at the core layer were upgraded without issue.

Test Procedure

The procedure used to perform the Upgrade of Supervisor 720 System in Core Layer (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Record the start time of this test using the **show clock** command.

Step 3 Verify that dca-core-1 is running the old Native Cisco IOS image using the **show version** command.

Step 4 Verify that the Supervisor 720 image under test is on the proper file device on dca-core-1 using the **dir disk0:** command.

Step 5 Use the **show running-config | include boot** command to verify that the boot string points to the proper device and filename for the test image. If any changes are necessary, make them and then save them to NVRAM when done.

Step 6 Issue the **reload** command on dca-core-1, causing the supervisor to reboot. Report any error messages seen during reload.

Step 7 Use the **show module** and **show version** commands to verify that dca-core-1 came online successfully and that the new image is running.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the upgrade process from SXF to SXF on the Supervisor 720 platform will proceed smoothly and without error.

Results

Upgrade of Supervisor 720 System in Core Layer (CSM Setup) passed.

Upgrade of Catalyst 4948-10GE System in Access Layer (CSM Setup)

This test verified the ability for code to be upgraded to the version of code that is under test. Access layer device dca-acc-4k-1 was upgraded from 12.2(31)SXG Native IOS to 12.2(31)SGA3 Native IOS to ensure that all hardware and configurations at the core layer were upgraded without issue.

Test Procedure

The procedure used to perform the Upgrade of Catalyst 4948-10GE System in Access Layer (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Record the start time of this test using the **show clock** command.

Step 3 Verify that dca-acc-4k-1 is running the old Native Cisco IOS image using the **show version** command.

Step 4 Verify that the image under test is on the proper file device on dca-acc-4k-1 using the **dir bootflash:** command.

The new image needs to be the first image on the bootflash: device in order for it to boot. The 4948-10GE system will boot the first image in bootflash:.

Step 5 Issue the **reload** command on dca-acc-4k-1, causing the system to reboot. Report any error messages seen during reload.

Step 6 Use the **show module** and **show version** commands to verify that dca-acc-4k-1 came online successfully and that the new image is running.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the upgrade process from 12.2(25)SG Native IOS to 12.2(31)SXG Native IOS on the 4948-10GE platform will proceed smoothly and without error.

Results

Upgrade of Catalyst 4948-10GE System in Access Layer (CSM Setup) passed.

Security

Resistence to outside attacks is critical to the operation of any data center. This section includes tests that measure the response of the network devices to various common attacks and techniques.

Secure Shell (SSH) is an application and a protocol that provide a secure replacement to the Berkeley r-tools. The protocol secures the sessions using standard cryptographic mechanisms, and the application can be used similarly to the Berkeley rexec and rsh tools. There are currently two versions of SSH available: SSH Version 1 and SSH Version 2.

The SSH Server feature enables a SSH client to make a secure, encrypted connection to a Cisco router. This connection provides functionality that is similar to that of an inbound Telnet connection. Before SSH, security was limited to Telnet security. SSH allows a strong encryption to be used with the Cisco IOS software authentication. The SSH server in Cisco IOS software will work with publicly and commercially available SSH clients.

The user authentication mechanisms supported for SSH are RADIUS, TACACS+ and the use of locally stored user names and passwords.

This section contains the following topics:

Malformed SNMP Polling

Each network device in the Data Center test topology is configured for both read-only and read-write access via SNMP. The availability of SNMP access of certain network devices to the outside world leaves them vulnerable to certain attacks. One possible attack is through the use of malformed SNMP packets.

This test relies on the Protos (http://www.ee.oulu.fi/research/ouspg/protos/) test suite for SNMP. This test application subjects the DUT to many hundreds of misconfigured SNMP packets in an attempt to disrupt system activity. The Protos SNMP test was run against device dca-agg-1 while that device was being monitored for errors and disruptions to CPU and memory stability.

Test Procedure

The procedure used to perform the Malformed SNMP Polling test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 If the background test traffic is not already running, start it now.

Step 3 Verify the SNMP community string settings default using the **show running-config | include snmp** command on dca-agg-1.

The read-only password is public (default).

Step 4 Execute the two Protos traffic generation scripts on dca-agg-1.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all DUT's not to pause indefinitely, crash, or give any tracebacks while test is being run.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Malformed SNMP Polling passed.

Malformed SSH Packets (CSM Setup)

Similar to its vulnerability to outside attacks via corrupt SNMP traffic, a network device may be susceptible to outside attacks via corrupt SSH traffic. This test relies on the Protos (http://www.ee.oulu.fi/research/ouspg/protos/) test suite for SSH. This test application subjects the DUT to many hundreds of misconfigured SSH packets in an attempt to disrupt system activity.

The Protos SSH test was run against the data center test network device dca-agg-1 while that device was being monitored for errors and disruptions to CPU and memory stability.

Test Procedure

The procedure used to perform the Malformed SSH Packets (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 If the background test traffic is not already running, start it now.

Step 3 Verify that dca-agg-1 is configured with a hostname, domain name, and TACACS authentication on the VTY lines using the following commands:

- show running-config | include hostname|domain|aaa|tacacs
- show running-config | begin line vty 0

The lines that should be present are as follows:

```
 hostname dca-agg-1 aaa new-model aaa authentication login default group tacacs+ local aaa
authorization exec default group tacacs+ if-authenticated local  aaa session-id common ip
domain-name example.com tacacs-server host 172.18.177.132 tacacs-server host
172.18.179.180 tacacs-server directed-request tacacs-server key cisco line vty 0 4
transport input telnet ssh
```

Step 4 Verify the SSH server on dca-agg-1 is enabled using the **show ip ssh** command and that dca-agg-1 is accepting SSH connections.

Step 5 Send malformed SSH packets to the device while monitoring the device. Ensure that the device does not pause indefinitely, crash, or reload.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect SSH vulnerability testing not to cause the router to reload, pause indefinitely, or crash.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Malformed SSH Packets (CSM Setup) passed.

NMAP Open Port Scan

A common way for hackers to wreak havoc on a network is to scan a network device (or an endpoint) for open TCP or UDP ports using the freely available NMAP tool. If an open port is found, the hacker may be able to exploit it and disrupt system activity. It is important, therefore, that a network device leave only those ports open that need to be for normal network services.

The test devices in the Data Center test topology have certain ports open by design. These include Telnet (port 23) and SSH (22). This test runs the NMAP Port scan tool against each device in the test topology, verifying that no ports open other than the ones expected. The DUT's are monitored for errors and CPU and memory stability during this procedure.

Test Procedure

The procedure used to perform the NMAP Open Port Scan test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin a port span on the Supervisor 720 devices in the test bed using the NMAP tool.

The command, run as root, that was used to execute this step was **nmap -v -p 1-65535***target_ip*.

Step 3 Verify that all open ports (as revealed by the port scan) are expected.

Each of the devices in the data center test topology have Telnet (TCP port 23) and SSH (TCP 22) open. These are the only ports we expect to see open.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the open ports revealed by the NMAP tool to be expected.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

NMAP Open Port Scan passed.

Traffic Forwarding

This section of tests measures some of the basic traffic forwarding features and abilities of the DCAP test topology.

This section contains the following topics:

Scaled FIB Consistency (CSM Setup)

Hardware forwarding in the Catalyst 6500 is accomplished by providing a specialized forwarding ASIC with a copy of the switch routing table. This Forwarding Information Base (FIB), located on the PFC3 of the Supervisor 720 engine, contains only the information from the routing table that is necessary for making a forwarding decision. This information includes the network prefix of the route, the next-hop address, and the egress interface. Because this FIB is located on the Supervisor 720 engine itself, the traffic must go here to be forwarded. This type of hardware forwarding is referred to as centralized.

The Catalyst 6500 switch family also allows for distributed forwarding in hardware through the use of Distributed Forwarding Cards (DFC's). These daughter cards, which in the Data Center topology are located on the WS-X6708-10GE and WS-X6704-10GE modules in the Aggregation layer, are equipped with their own FIB, which is also a forwarding ASIC. This distributed FIB is synchronized with the FIB residing on the PFC3. The end result is faster forwarding, because forwarding lookups can be done locally on the line card and are not needed from the supervisor engine.

This test verified that the FIBs on the WS-X6708-10GE and WS-X6704-10GE line cards in the Aggregation Layer are properly synchronized. Device dcb-agg-1 has seven WS-X6708-10GE linecards installed. Device dcb-agg-2 has seven WS-X6704-10GE linecards installed. In each aggregation device, dcb-agg-1 and dcb-agg-2, the central FIB is inspected and compared to each of the five distributed FIB's. The devices under test were monitored for errors and CPU and memory utilization issues.

Test Procedure

The procedure used to perform the Scaled FIB Consistency (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Log into dcb-agg-1 and use the **show module** command to verify the location of any DFC's in the system.

There are DFC's in each of slots 1-4 and 6-8 in dcb-agg-1.

Step 3 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 1.

Step 4 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 2.

Step 5 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 3.

Step 6 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 4.

Step 7 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 6.

Step 8 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 7.

Step 9 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 8.

Step 10 Log into dcb-agg-2 and use the **show module** command to verify the location of any DFC's in the system.

There are DFC's in each of slots 1-4 and 6-8 in dcb-agg-2.

Step 11 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 1.

Step 12 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 2.

Step 13 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 3.

Step 14 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 4.

Step 15 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 6.

Step 16 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 7.

Step 17 Use the **show ip cef** command to verify that there is no difference between the FIB on the PFC and the FIB on the DFC in slot 8.

Step 18 Stop background scripts to collect final status of network devices and analyze for error.

Step 19 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect there to be total parity between the FIB maintained on the supervisor (PFC) and the FIB's maintained on the DFC's.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Scaled FIB Consistency (CSM Setup) passed.

Zero Packet Loss (CSM Setup)

This test verified that the network devices in the Data Center topology are able to forward basic network traffic, without loss, in a steady-state condition. Web (HTTP/HTTPS) traffic consisting of varying frame sizes is sent between client devices and web servers. No negative, or failure, events are introduced during this test. The network devices will all be monitored for errors, and for CPU and memory usage stability.

Test Procedure

The procedure used to perform the Zero Packet Loss (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin the background test traffic that will send 30 minutes' worth of HTTP, HTTPS, and FTP traffic between the clients and the servers.

Step 3 When the traffic completes, measure the percentage of connection attempts that resulted in error codes. This percentage should be less than one percent.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss experienced by the background test traffic to be within tolerances.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Zero Packet Loss (CSM Setup) passed.

Layer 2 Protocols

This section of the test plan looks at the functionality of some of the common Layer 2 protocols used in the test topology. This encompasses layer 2 of the protocol stack.

This section contains the following topics:

Link Aggregation Control Protocol (LACP)

There are several ways that a channel can be formed using the LACP protocol. The channel that is used in the Data Center test topology is configured using LACP active mode, in which the port initiates negotiations with other ports by sending LACP packets.

This section contains the following topics:

LACP Basic Functionality

There are several ways that a channel can be formed using the LACP protocol. The channel that is used in the Data Center test topology is configured using LACP active mode, in which the port initiates negotiations with other ports by sending LACP packets. This test verified that the channel is formed correctly between dca-agg-1 and dca-agg-2. The CPU and memory utilization are monitored for stability during this test.

Test Procedure

The procedure used to perform the LACP Basic Functionality test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On both dca-agg-1 and dca-agg-2, it is interfaces TenGigabit Ethernet 9/3 and TenGigabit Ethernet 10/3 that are bundled to form Port-channel 1 using LACP. Use the **show running-config interface** command on each of these interfaces to verify that LACP is configured for active mode.

The following lines are present on each of these four interfaces:

```
channel-protocol lacpchannel-group 1 mode active
```

Step 3 Use the **show interfaces Port-channel 1 etherchannel** command on both dca-agg-1 and dca-agg-2 to verify that both interfaces Te9/3 and Te10/3 are bundled and active in the port-channel.

The "Number of ports" in each case should be given as "2". Further, each of the two interfaces should be listed as "Ports in the Port-channel" and their "EC state" should be "Active".

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the LACP-formed channels will build correctly.
- We expect no CPU or memory problems.

Results

LACP Basic Functionality passed.

LACP Load Balancing (CSM Setup)

When the next-hop for network traffic is out an etherchannel, the switch must decide which of the bundled physical interfaces to send the network traffic out. Further, the switch must have the ability to balance any traffic going out an etherchannel across they multiple available physical interfaces (anything less would be a waste of available bandwidth). In Native IOS, there are several etherchannel load-balancing algorithms available for the network administrator to use to get the best balance of traffic across all available physical interfaces.

The algorithm used in the Data Center test topology makes the load balancing decision (which physical port to send the traffic out) based on a combination of the source and destination Layer 4 ports. This test verified that both physical interfaces in the etherchannel between dca-agg-1 and dca-agg-2 passed traffic when a diversity of traffic was sent across the etherchannel. The Aggregation layer devices were monitored for any errors. The CPU and memory utilization were monitored for stability.

Test Procedure

The procedure used to perform the LACP Load Balancing (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Use the **show interfaces Port-channel 1 etherchannel** command on dca-agg-1 and dca-agg-2 to verify that a two-port channel is active between the two devices.

The channel shows ports Te9/3 and Te10/3 in Active state.

Step 3 Use the **show running-config | include load-balance** command to verify that dca-agg-1 and dca-agg-2 are configured to do Layer 4 source/destination load-balancing.

The configuration command **port-channel load-balance src-dst-port** is present on both devices.

Step 4 Clear the traffic counters on dca-agg-1 and dca-agg-2 using the **clear counters** command.

Step 5 Begin a 5-minute period of the background test traffic.

Step 6 When the traffic has finished, use the **show interfaces Port-channel 1 counters etherchannel** command on dca-agg-1 and dca-agg-2 to verify that traffic was sent on both ports of the etherchannel.

The distribution of traffic may or may not be equal, depending on the distribution of source and destination ports for ingress and egress traffic.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that traffic will be distributed between both links of the etherchannel connecting dca-agg-1 and dca-agg-2.
- We expect no CPU or memory problems.

Results

LACP Load Balancing (CSM Setup) passed.

Spanning-Tree Protocol (STP)

The IEEE 802.1d Spanning Tree specification allows physical path redundancy without active network "loops" by defining a tree that spans all of the switches in an extended network and then forces certain redundant data paths into a standby (blocked) state. At regular intervals, the switches in the network send and receive spanning tree packets that they use to identify the path. If one network segment becomes unreachable, or if spanning tree costs change, the spanning tree algorithm reconfigures the spanning tree topology and reestablishes the link by activating the standby path.

Each of the seven devices in the topology Layer 2 domain participates in Spanning-Tree. The Spanning-Tree Protocol (STP) that is used in the DCAP topology is PVST+ plus the rapid convergence enhancements of IEEE 802.1w (collectively referred to as Rapid PVST+ or rPVST+). This group of tests looks at the basic functionality of rPVST+ as well as some of the commonly-used STP features.

This section contains the following topics:

Rapid PVST+ Basic Functionality (CSM Setup)

In the Data Center test topology dca-agg-1 is configured to be the primary root switch for all VLANs, which dca-agg-2 is configured to be the secondary root switch. This test does not focus so much on the ability of rPVST+ to converge quickly and accurately as it does on the fundamental mechanics of STP. It verifies that the correct switch is root and that all Layer 2 interfaces in the Data Center Layer 2 domain are in the correct STP state, with the correct switch identified as root, for all VLANs.

Test Procedure

The procedure used to perform the Rapid PVST+ Basic Functionality (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that spanning-tree configurations on all Layer 2 devices in the DCAP test topology using the **show running-configuration | include spanning-tree**.

On devices dca-agg-1, dca-agg-2, dca-acc-6k-1, dca-acc-6k-2, dca-acc-4k-1, dca-acc-4k-2 and dca-voodoo-2, the following lines are present:

- **spanning-tree mode rapid-pvst**
- **spanning-tree extend system-id**
- **spanning-tree pathcost method long**

On dca-agg-1 (which is configured as the root switch for all VLANs) the following configuration line should be present:

- **spanning-tree vlan 1-4094 priority 24576**

On dca-agg-2 (which is configured as the secondary root switch for all VLANs) the following configuration line should be present:

- **spanning-tree vlan 1-4094 priority 28672**

Step 3 Verify that the system with MAC address 0015.c719.bf80 is root for all VLANs on all systems using the **show spanning-tree root** command.

Note that a different system may be shown as root for VLAN 1 on some systems. This is expected as VLAN 1 is active only because it is allowed on the CSM port-channel (Po258) in both dca-agg-1 and dca-agg-2. For this reason, each of those two devices will report their respective local MAC addresses as being root for VLAN 1.

Step 4 Verify that dca-agg-1 is the system that owns the root MAC address 0015.c719.bf80 using the **show catalyst6000 chassis-mac-addresses** command.

Note that on the Catalyst 4900 systems, dca-acc-4k-1 and dca-acc-4k-2, the **show module** command will be used to verify that this root MAC address does not fall into the range of system MAC addresses.

Step 5 Use the **show spanning-tree vlan 2101** command to map the spanning-tree for VLAN 2101 as an example of what the per-VLAN STP topology should look like.

The device dca-agg-1, which is the STP root for VLAN 2101, should report all interfaces in "FWD" state. This list of interfaces includes Te9/4, Te10/1, Te10/2, Te10/4, Po1 and Po270 (the FWSM/backplane interface). The Root ID Address should show the root MAC address "0015.c719.bf80" as should the Bridge ID Address (this switch is root). The Root ID Priority and the Bridge ID Priority should also be the same value, "25677", which is the configured priority of "24576" plus the VLAN, 2101.

The device dca-agg-2, which is the secondary STP root for VLAN 2101, should report all interfaces in "FWD" state. This list of interfaces includes Te9/4, Te10/1, Te10/2, Te10/4, Po1 and Po270 (the FWSM/backplane interface). The Root ID Address should show the root MAC address "0015.c719.bf80". The Bridge ID Address should show the local system MAC address "0015.c734.9d80". The Root ID Priority should be "25677", while the Bridge ID Priority should be "30773", which is the configured priority of 28672 plus the VLAN, 2101.

The device dca-acc-6k-1, should report interface Te1/1 in "FWD" state and Te1/2 in "BLK" state. All other interfaces (connected to the servers in the DCAP test topology) should be in "FWD" state. The Root ID Address should show the root MAC address "0015.c719.bf80". The Bridge ID Address should show the local system MAC address "0016.9cb5.c000". The Root ID Priority should be "25677", while the Bridge ID Priority should be "34869", which is the default priority of 32768 plus the VLAN, 2101.

The device dca-acc-6k-2, should report interface Te1/1 in "FWD" state and Te1/2 in "BLK" state. All other interfaces (connected to the servers in the DCAP test topology) should be in "FWD" state. The Root ID Address should show the root MAC address "0015.c719.bf80". The Bridge ID Address should show the local system MAC address "0016.9c9e.a000". The Root ID Priority should be "25677", while the Bridge ID Priority should be "34869", which is the default priority of 32768 plus the VLAN, 2101.

The device dca-acc-4k-1, should report both interfaces Te1/49 and Te1/50 in "FWD" state. All other interfaces (connected to the servers in the DCAP test topology) should also be in "FWD" state. The Root ID Address should show the root MAC address "0015.c719.bf80". The Bridge ID Address should show the local system MAC address "0015.fa80.4f80". The Root ID Priority should be "25677", while the Bridge ID Priority should be "34869", which is the default priority of 32768 plus the VLAN, 2101.

The device dca-acc-4k-2, should report interface Te1/49 in "FWD" state and Te1/50 in "BLK" state. All other interfaces (connected to the servers in the DCAP test topology) should be in "FWD" state. The Root ID Address should show the root MAC address "0015.c719.bf80". The Bridge ID Address should show the local system MAC address "0015.fa80.4f40". The Root ID Priority should be "25677", while the Bridge ID Priority should be "34869", which is the default priority of 32768 plus the VLAN, 2101.

The device dca-voodoo-2, should report interface Te1/1 in "FWD" state and Te1/2 in "BLK" state. All other interfaces (connected to the servers in the DCAP test topology) should be in "FWD" state. The Root ID Address should show the root MAC address "0015.c719.bf80". The Bridge ID Address should show the local system MAC address "000f.f827.4d80". The Root ID Priority should be "25677", while the Bridge ID Priority should be "34869", which is the default priority of 32768 plus the VLAN, 2101.

Step 6 Use the **show spanning-tree summary** command to verify that all VLANs on the primary root (dca-agg-1) and secondary root (dca-agg-2) are in "Forwarding" state.

The very last line of the output for this command gives a summary for all VLANs. There should be no VLANs in any state other than "Forwarding".

Step 7 Use the **show spanning-tree summary** command to verify that all VLANs on the access switches (dca-acc-6k-1, dca-acc-6k-2, dca-acc-4k-1, dca-acc-4k-2 and dca-voodoo-2) have a single port in "Blocking" state (with the exception of dca-acc-4k-1, which will have all "Forwarding").

In each VLAN row, in the output of this command, there should be a "1", indicating a single port is "Blocking" for that VLAN.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the spanning-trees for each VLAN will be converged correctly, with the appropriate ports in the appropriate STP state.
- We expect no CPU or memory problems.

Results

Rapid PVST+ Basic Functionality (CSM Setup) passed.

Root Guard

Spanning-tree best practices dictate that the root switch (and even backup root switch) should be designated and forced into that role by the network designer. This is done by configuring the STP priority to an appropriate value lower than the default of 32768. A deterministic root switch results in deterministic network traffic and predictable behavior.

This predictable behavior can be disrupted, however, should a switch be inserted into the network topology (accidentally or maliciously) with a lower STP priority or bridge ID than the configured root switch. The STP Root Guard feature helps to protect against such a situation by building a wall of protection around the configured root switches.

Interfaces that are connecting the root switches to the access layer switches are configured locally with the Root Guard feature. Should the root (or secondary root) receive a BPDU on any of these interfaces with a lower bridge ID than it has, the interface will be transitioned into a Root Inconsistent state. This is essentially a perpetual Listening state in which the interface can continue to monitor the link for errant BPDU's (or their absence), but not forward any data traffic. When the interface stops receiving such BPDU's, it transitions back to the appropriate STP state.

In the DCAP test topology, dca-agg-1 is configured to be the primary STP root while dca-agg-2 is configured to be the secondary STP root. The interfaces that connect these two switches to the Layer 2 access devices are configured with STP Root Guard enabled. The port-channel interface connecting these two aggregation devices have Root Guard disabled.

In this test, Port-channel 1 (connecting dca-agg-1 and dca-agg-2) will be broken. When this happens, the interfaces connecting the access switches to dca-agg-2 will transition from Blocking to Forwarding state and begin to forward BPDU's from dca-agg-1 to dca-agg-2. The device dca-agg-2, now receiving BPDU's of a lower priority (from the STP root), will move the links on which it is receiving such BPDU's to Root Inconsistent state. When Port-channel 1 is reconnected, the links will return to Forwarding state.

Test Procedure

The procedure used to perform the Root Guard test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Use the **show running-config | include spanning-tree** command to verify that dca-agg-1 is configured to be the primary STP root switch and dca-agg-2 is configured to be the secondary root switch.

The device dca-agg-1 is configured with a priority of 24576, the lowest of any switches in the DCAP test topology. It will therefore assume the primary STP root role. The device dca-agg-2 is configured with a priority of 28672, the lowest of any switches in the DCAP test topology. It will therefore assume the secondary STP root role.

Step 3 Use the **show running-config interface***interface* command to verify that interfaces Te9/4, Te10/1, Te10/2, and Te10/4 on both dca-agg-1 and dca-agg-2 are configured with **spanning-tree guard root**.

Step 4 Use the **show spanning-tree interface***interface* command to verify that interfaces Te9/4, Te10/1, Te10/2, and Te10/4 on both dca-agg-1 and dca-agg-2 are in STP Forwarding state for all VLAN's.

Step 5 Verify that there are no interfaces in Root Inconsistent state on either dca-agg-1 or dca-agg-2 using the **show spanning-tree inconsistentports** command.

Step 6 Shutdown Port-channel 1 on dca-agg-2.

Step 7 Verify that Te9/4, Te10/1, Te10/2, and Te10/4 on dca-agg-2 are in Root Inconsistent state using the **show spanning-tree inconsistentports** command.

Each of these interfaces should be listed per VLAN in the output of this command.

Step 8 Bring Port-channel 1 on dca-agg-2 back online using the **no shutdown** command.

Step 9 Use the **show spanning-tree interface***interface* command to verify that interfaces Te9/4, Te10/1, Te10/2, and Te10/4 on both dca-agg-1 and dca-agg-2 are again in STP Forwarding state for all VLAN's.

Step 10 Verify that there are again no interfaces in Root Inconsistent state on either dca-agg-1 or dca-agg-2 using the **show spanning-tree inconsistentports** command.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the interfaces with STP root guard enabled to transition to Root Inconsistent state when they begin receiving BPDU's with lower priority than the local BPDU.
- We expect the interfaces with STP root guard enabled to return to Forwarding state when they stop receiving such BPDU's.
- We expect no CPU or memory problems.

Results

Root Guard passed.

Trunking

A trunk is a point-to-point link between one or more switch ports and another networking device such as a router or a switch. Trunks carry the traffic of multiple VLANs over a single link and allow VLANs to be extended across an entire network. The table lists and describes the five modes of trunking on Cisco switches.

Mode	Description
On	Local interface trunks. Sends Dynamic Trunking Protocol (DTP) packets. Puts the port into permanent trunking mode and negotiates to convert the link to a trunk link. The port becomes a trunk port even if the neighboring port does not agree to the change.
Off	Local interface does not trunk. Puts the port into nontrunking mode and negotiates to convert the link into a nontrunk link. The port becomes a nontrunk port even if the neighboring port does not agree to the change.
Auto	Local interface trunks if it receives DTP packets. Enables the port to convert the link to a trunk link. The port becomes a trunk port if the neighboring port is set to on or desirable mode. This is the default mode for Fast Ethernet and Gigabit Ethernet ports.
Desireable	Local interface sends DTP packets. Makes the port actively attempt to convert the link to a trunk line. The port becomes a trunk port if the neighboring port is set to on, desirable, or auto mode.
Nonegotiate	Local interface forms a trunk and does not send DTP packets. Puts the port into permanent trunking mode, but prevents the port from generating DTP frames. You must configure the neighboring port normally as a trunk port to establish a trunk link.

This section contains the following topics:

- 802.1q Trunking Basic Functionality, page 2-34

802.1q Trunking Basic Functionality

On Cisco Catalyst 6500 and Catalyst 4900 switches, trunks can be formed in multiple ways. Trunking can either be dynamic, in which trunking is negotiated between the two sides of the link, or it can be set to **on** or **off**, statically. In the case of the Data Center test topology, the trunk links are set to **on**, meaning that they will trunk VLANs regardless of what the remote side of the link is doing.

The trunk encapsulation can also be either dynamically negotiated or set statically. In the Data Center test topology, the encapsulation is set statically to 802.1q, or **dot1q**.

This test verified that the links that are configured as trunk links between the Data Center devices actually form trunks correctly. The links looked at include those between two Catalyst 6500s (dca-agg-2 and dca-acc-6k-1) and those between a Catalyst 6500 and a Catalyst 4900 (dca-agg-2 and dca-acc-4k-2). The CPU and memory utilization of the DUTs was monitored for stability.

Test Procedure

The procedure used to perform the 802.1q Trunking Basic Functionality test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 The devices dca-agg-2 and dca-acc-6k-1, both Catalyst 6500s are connected by a static trunk. Use the **show running-config interface***interface* and **show interfaces***interface***trunk** commands to verify that this is the current configuration and the trunk is currently working.

Step 3 Using the **shutdown** and **no shutdown** commands, flap the Te9/4 interface on dca-agg-2.

Step 4 Use the **show interfaces***interface***trunk** command to verify that the trunk between dca-agg-2 and dca-acc-6k-1 has re-formed correctly.

Step 5 The devices dca-agg-2 (a Catalyst 6500) and dca-acc-4k-2 (a Catalyst 4900) are connected by a static trunk. Use the **show running-config interface***interface* and **show interfaces***interface***trunk** commands to verify that this is the current configuration and the trunk is currently working.

Step 6 Using the **shutdown** and **no shutdown** commands, flap the Te10/2 interface on dca-agg-2

Step 7 Use the **show interfaces***interface***trunk** command to verify that the trunk between dca-agg-2 and dca-acc-4k-2 has re-formed correctly.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the 802.1q trunks will be formed correctly between the two Catalyst 6500 devices.
- We expect that the 802.1q trunks will be formed correctly between the Catalyst 6500 and the Catalyst 4900.
- We expect no CPU or memory problems.

Results

802.1q Trunking Basic Functionality passed.

Unidirectional Link Detection (UDLD)

The Unidirectional Link Detection (UDLD) protocol allows devices connected through fiber-optic or copper Ethernet cables (for example, Category 5 cabling) to monitor the physical status of the cables and detect when a unidirectional link exists. When a unidirectional link is detected, UDLD shuts down the affected port and alerts the user. Unidirectional links can cause a variety of problems, including spanning-tree topology loops and erroneous Layer 3 routing.

This section contains the following topics:

UDLD Detection on 10-Gigabit Ethernet Links

This test forced a unidirectional link condition on one of the Ten Gigabit Ethernet links in the Data Center test topology and verified that the link was put into a UDLD down state correctly.

Devices dca-agg-1 and dca-agg-2 are connected via two TenGigabit Ethernet links, Te9/3 and Te10/3 (on each device). On dca-agg-1, the tx fibers of Te9/3 and Te10/3 were switched, creating a crossed-fiber situation.

The DUTs were monitored for errors during this test. The CPU and memory utilization on the DUTs were also monitored for stability.

Test Procedure

The procedure used to perform the UDLD Detection on 10-Gigabit Ethernet Links test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Use the **show udld***interface* command to verify the current UDLD state of interfaces Te9/3 and Te10/3 on both dca-agg-1 and dca-agg-2.

The operational state for all interfaces should be "Enabled / in aggressive mode". The current bidirectional state should be "Bidirectional". There should also be a single neighbor entry showing "Bidirectional" as the current bidirectional state.

Step 3 Switch the transmit fibers of interfaces Te9/3 and Te10/3 on dca-agg-1. Verify that the system log contains at least one UDLD link detection interface disable message.

Step 4 Verify the interface status for all four interfaces using the **show interface***interface***status** command.

check for the errdisable state.

Step 5 Use the **show udld***interface* command to verify the current UDLD state of interfaces Te9/3 and Te10/3 on dca-agg-1 and dca-agg-2.

Step 6 Return the transmit fibers to their original location and flap interface Te10/3 on dca-agg-1 and dca-agg-2 using the **shutdown** and **no shutdown** commands.

Step 7 Use the **show udld***interface* command to verify that interfaces Te9/3 and Te10/3 on both dca-agg-1 and dca-agg-2 have returned to the original UDLD states.

The operational state for all interfaces should be "Enabled / in aggressive mode". The current bidirectional state should be "Bidirectional". There should also be a single neighbor entry showing "Bidirectional" as the current bidirectional state.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect one or more switches to have ports in errdisable state when UDLD determines a fiber has been crossed between ports.
- We expect UDLD Link State to be shutdown for the port determined to be unidirectional.
- We expect no CPU or memory problems.

Results

UDLD Detection on 10-Gigabit Ethernet Links passed.

Layer 3 Protocols

This section of the test plan looks at the functionality of some of the common Layer 3 protocols used in the DCAP test topology.

This section contains the following topics:

Hot Standby Router Protocol (HSRP)

Hot Standby Router Protocol (HSRP) is used to provide a redundant gateway IP address to clients on a particular subnet. In the DCAP test topology, the virtual IP address (gateway) is shared by two routers, dca-agg-1 and dca-agg-2. Each of these two routers is configured with two IP addresses per HSRP subnet, one that is unique to that router, and one that is shared with the peer HSRP router. The router with the highest HSRP priority will assume Active state and respond to queries on the Virtual IP. The other router will assume Standby state and ignore such queries, while in Standby state.

This section contains the following topics:

HSRP Basic Functionality

Hot Standby Router Protocol (HSRP) is used to provide a redundant gateway IP address to clients on a particular subnet. In the DCAP test topology, the virtual IP address (gateway) is shared by two routers, dca-agg-1 and dca-agg-2. Each of these two routers is configured with two IP addresses per HSRP subnet, one that is unique to that router, and one that is shared with the peer HSRP router. The router with the higher HSRP priority will assume Active state and respond to queries on the Virtual IP. The other router will assume Standby state and ignore such queries while in Standby state.

There are 200 HSRP groups in the DCAP test topology, providing virtual gateways for over 200 subnets. This test verified that the Aggregation Layer devices were able to scale to this number of standby groups. It verified that the correct router was in Active HSRP state and that only that router was displaying the HSRP MAC address.

Test Procedure

The procedure used to perform the HSRP Basic Functionality test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the HSRP configuration for VLAN's 1101 to 1300 on dca-agg-1 and dca-agg-2 using the **show running-config | begin interface Vlan1101** command.

On dca-agg-1, these 200 VLAN's are configured with a standby priority of 120. On dca-agg-2, they are configured with a standby priority of 110. Each VLAN has a standby IP address, and each belongs to a separate standby group (specified by the number directly following standby).

Step 3 Use the **show standby brief** command to verify that dca-agg-1 is active for VLAN's 1101 to 1300 and that dca-agg-2 is standby.

Step 4 Verify that dca-agg-1 has a virtual MAC address running on each of the standby VLAN's for which it is the active HSRP router using the **show standby | include Vlan|Virtual mac**.

Each VLAN has a virtual MAC address assigned to it.

Step 5 Verify that dca-agg-2 does not have a virtual MAC address running on the standby VLAN's for which it is the standby HSRP router using the **show standby | include Vlan|Virtual mac**.

None of the VLAN's have a virtual MAC address assigned to it.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all HSRP groups to reflect their configuration in their active and standby states.
- We expect all active HSRP groups to have an associated virtual MAC address and that no standby HSRP groups will have a MAC address.
- We expect no CPU or memory problems.

Results

HSRP Basic Functionality passed.

Open Shortest Path First (OSPF)

Open Shortest Path First (OSPF) is an Interior Gateway Protocol (IGP) developed by the OSPF working group of the Internet Engineering Task Force (IETF). Designed expressly for IP networks, OSPF supports IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets.

This section contains the following topics:

OSPF Database Verification (CSM Setup)

Each Layer 3 device in an autonomous system running OSPF maintains a detailed view, or database, of the entire OSPF network. This database contains information gathered about networks that have been advertised via OSPF, and OSPF routers in the network. Information about networks and OSPF routers is propagated through the OSPF network using several different types of Link State Algorithm (LSA) messages.

This test verified that the OSPF database contains the information that would be expected for this particular test topology.

Test Procedure

The procedure used to perform the OSPF Database Verification (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 From each router in the DCAP test topology that is running OSPF, gather information about the OSPF interfaces and the OSPF processes that are running using the **show ip ospf database**, **show ip ospf database database-summary**, and **show ip ospf interfaces** commands.

The devices in the DCAP test topology that are running OSPF include dca-agg-1, dca-agg-2, dca-core-1, dca-core-2, and dca-voodoo-2.

Step 3 Verify that the number of Router (Type 1) LSA's in each area is what is expected.

For each router in a particular area, there should be a single router LSA given.

Step 4 Verify that the number of Network (Type 2) LSAs in each area is what is expected.

A Type 2 LSA is generated when a network segment exists that has both a DR and a BDR. In other words, when a network shared by two devices are both running OSPF on that network, a network LSA is generated. The number of Network LSA's is determined by how many of those types of links are in a given area.

Step 5 Verify that the number of Summary Network (Type 3) LSA's in each area is what is expected.

The ABR, or the router that is at the border of two or more areas, sends a summary network LSA into Area X for all other ares other than X for which it is a participant. So if you have Areas X, Y, and Z and RouterA is a participant in all of them, it will generate a Type-3 LSA and send it into Area X. This LSA will contain a summary of all the networks found in Areas Y and Z.

Step 6 Verify that the number of Summary ASBR (Type 4) LSAs in each area is what is expected.

An ABR between areas X and Y will generate a Summary ASBR into X for an ASBR that exists in Y. However, an ABR which is an ASBR will not advertise itself. In each area, verify there is a Summary ASBR LSA from each ABR in that area for each ASBR outside this area.

Step 7 Verify that the number of AS-External (Type 5) LSA's in each area is what is expected.

This is pretty much the number of networks that are being redistributed from another routing protocol into OSPF times the number of ABR's that share redistribution responsibilities.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the OSPF database to be populated with the correct number of LSA's of each type for the topology being used.

Results

OSPF Database Verification (CSM Setup) passed.

OSPF Pagent Convergence Test Second Aggregate Layer

The aim of this test is to inject routes into the core layer, simulating a second aggregate layer. The test then uses Pagent's Convergence Test to remove and re-insert routes into the core layers tocheck the routes are correctly propagated through the network topology

Test Procedure

The procedure used to perform the OSPF Pagent Convergence Test Second Aggregate Layer test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Ensure that the two interfaces connecting to the aggregation layer are shut down on the pagent, by using the following commands:

fastEthernet 2/0 off fastEthernet 2/1 off Enable the pagent connections to the Core layer with the following commands

fastEthernet 1/0 on fastEthernet 1/1 on

Step 3 Begin LNE OSPF using the following commands:

start

Step 4 Verify on each core device that the Pagent Box has successfully neighboured via OSPF, using the following command (on core 1 and core 2):

show ip ospf neighbor

Step 5 Verify that all the routes have been advertised correctly to Core 1, use the following command:

sh ip route

Check that the routes to each subnet are being learnt via the connecting interface to the pagent, and via the connected core on TenGig 1/1.

Step 6 Verify that all the routes have been advertised correctly to Core 2, use the following command:

show ip route

Check that the routes to each subnet are being learnt via the connecting interface to the pagent, and via the connected core on TenGig 1/1.

Step 7 Check the routes are summarised at the client switch:

show ip route

Step 8 Clear the convergence test statistics by issuing the **clear convergence-stats** command.

Step 9 Configure the convergence test to perform 20 withdraw/ restore iterations with a gap of 20 seconds between each iteration.

Step 10 Turn on verbose mode for the convergence test, and begin the test using the following commands:

convergence-test verbose on convergence-test on

Step 11 When the test has completed, you will see the following message, then enter the **show convergence-stats** command to display test results.

Check that the average convergence times are within a reasonable time frame.

Step 12 Issue the **stop** command on the Pagent router to stop the OSPF service. For each interface connected to a core or aggregation device, issue the command **off** to disable the interfaces.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that The convergence test will be able to remove/ insert the routes, and the topology will converge to reflect the change.
- We expect no CPU or memory problems.

Results

OSPF Pagent Convergence Test Second Aggregate Layer passed.

OSPF Pagent Router Flap Test Second Aggregate Layer

The aim of this test is to inject routes into the core layer, simulating a second aggregate layer. The test then uses Pagent's Router-Flap test to simulate an entire device shutting down, then coming back up.

Test Procedure

The procedure used to perform the OSPF Pagent Router Flap Test Second Aggregate Layer test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Ensure that the two interfaces connecting to the aggregation layer are shut down on the pagent, by using the following commands:

fastEthernet 2/0 off fastEthernet 2/1 off

Step 3 Configure FastEthernet1/1 to advertise the same routes as FastEthernet1/0, using the folowing command:

network start 110.40.0.0

Step 4 Begin LNE OSPF using the following commands: **start**

Step 5 Verify on each core device that the Pagent Box has successfully neighboured via OSPF with core 1 and core 2, by using the following command:

show ip ospf neighbor

Step 6 Verify that the 110.40.0.0 route is being installed from both connecting interfaces to the cores and on the client switch:

sh ip route 110.40.0.0

Step 7 Configure router flapping on the Pagent router, using the following commands:

FastEthernet1/0 router-flap duration on 60 to 120 seconds router-flap duration off 60 to 120 seconds router-flap on

This will configure the Pagent to simulate a router flap. It causes the OSPF process to be active for 60 seconds to 120 seconds, and then act as if it has been turned off for 60 to 120 seconds.

Note: You can use the show router-flap command (on the Pagent router) to display when the router will be flapped.

Step 8 Use the **show router-flap** command to check when the current state is set to on.

When you notice that the current state is set to on, use the **show ip route 110.40.0.0** on the client switch to verify that one of the summary routes has been removed from the routing table. Verify that this route is only learnt via the connection to core 2.

Step 9 Allow the router-flap to flap for 5 iterations, and stop the router flap using the **router-flap off** command. Afterwards check that the routes are still in the routing table (on client switch), use the **show ip route 110.40.0.0** command.

Step 10 Issue the **stop** command on the Pagent router to stop the OSPF service. For each interface connected to a core or aggregation device, issue the command **off** to disable the interfaces.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the router-flaps will be seen, and the network will converge accordingly, within the reasonable time frames.
- We expect no CPU or memory problems.

Results

OSPF Pagent Router Flap Test Second Aggregate Layer passed.

OSPF Pagent Verify Test Second Aggregate Layer

The aim of this test is to inject routes into the core layer, simulating a second aggregate layer. The test then uses Pagent's Verify Test to check the routes have been correctly propagated through the network topology via OSPF.

Test Procedure

The procedure used to perform the OSPF Pagent Verify Test Second Aggregate Layer test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Ensure that the two interfaces connecting to the aggregation layer are shut down on the pagent, by using the following commands:

fastEthernet 2/0 off fastEthernet 2/1 off

Step 3 Begin LNE OSPF using the following commands: **start**

Step 4 Verify on each core device that the Pagent Box has successfully neighboured via OSPF, using the following command: **show ip ospf neighbor**

Step 5 Verify that the summary route has sucessfully been propagated to the client switch using the following command (from the client switch):

show ip route

Step 6 Clear the verify test statistics by issuing the following command:

clear verify-stats

Step 7 Begin the Verify Test using the following command:

verify-test on

Step 8 Issue the show verify-stats command to confirm that all 100 installed routes have been verified. The Unreach and Unknown variables should be set to zero.

show verify-stats

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the Pagent device will verify the injected routes through the topology.
- We expect no CPU or memory problems.

Results

OSPF Pagent Verify Test Second Aggregate Layer passed.

OSPF Route Summarization (CSM Setup)

In the DCAP test topology, in OSPF Area 10, there are several /30 subnets configured for the inter-switch links. These all share a 172.31.1.x prefix. All servers in Area 10 are on /24 subnets and share a prefix of 101.1.x.x.

The Core routers in the Data Center test topology, dca-core-1 and dca-core-2, as part of the default configuration, summarize the subnets in Area 10 for advertisement into Area 0. The 172.31.1.x/30 subnets are configured to summarize as 172.31.1.0/24 networks while the 101.1.x.x/24 subnets are configured to summarize as 101.1.0.0/16 networks.

This test verified that summarization was occurring by looking at the routing table of a device in Area 20. The memory and CPU utilization were monitored for stability.

Test Procedure

The procedure used to perform the OSPF Route Summarization (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Use the **show running-config | begin router ospf** command on dca-core-1 and dca-core-2 to verify that summarization is configured.

Each of the devices will have the following two lines as part of their OSPF configuration:

area 10 range 101.1.0.0 255.255.0.0area 10 range 172.31.1.0 255.255.255.0

The first line will cause any networks in Area 10 matching 101.1.x.x to be summarized into a /16. The second line will cause any networks in Area 10 matching 172.31.1.x to be summarized into a /24.

Step 3 On dca-agg-1, use the **show running-config interface Te9/2** to verify that this device has an interface with a /30 address matching the 172.31.1.x format.

The IP address of interface Te9/2 is 172.31.1.14/30.

Step 4 On dca-voodoo-2, an Area Border Router in Area 20, use the **show ip route 172.31.1.14** command to verify that the route to this address has been summarized as a /24.

The output of this command will show a "Routing entry for 172.31.1.0/24".

Step 5 On dca-agg-1, use the **show running-config interface Vlan1101** to verify that this device has an interface with a /24 address matching the 101.1.x.x format.

The IP address of interface Vlan1101 is 101.1.1.2/24.

Step 6 On dca-voodoo-2, use the **show ip route 101.1.1.2** command to verify that the route to this address has been summarized as a /16.

The output of this command will show a "Routing entry for 101.1.0.0/16".

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the appropriate networks will be summarized correctly by the OSPF Area Border Routers.
- We expect no CPU or memory problems.

Results

OSPF Route Summarization (CSM Setup) passed.

Negative

The suite of negative tests are aimed at measuring the response of the test topology to various negative events, such as simulated hardware failures, link failures and whole device failures.

This section contains the following topics:

Hardware Failure

This group of tests measures the ability of the test topology to absorb various hardware failures, including system crashes and module resets.

This section contains the following topics:

Access Layer Supervisor Failover Using SSO with NSF (CSM Setup)

Of the failover protocols that Cisco Catalyst 6500 series switches support, SSO is the most aggressive and provides the shortest downtimes. With the Stateful Switchover protocol, the standby supervisor is fully booted and ready, with a copy of the synchronized configuration received from the active supervisor.

Coupled with the Non-Stop Forwarding feature, which allows the forwarding of traffic even while forwarding tables are being rebuilt by the new supervisor, SSO has the ability to provide sub-second failovers.

In the DCAP test topology, the only devices with supervisor redundancy are the Catalyst 6500 access switches, dca-acc-6k-1 and dca-acc-6k-2. This test measures the effect of an SSO/NSF failover on connections facilitated by dca-acc-6k-2. A series of ten SSO/NSF failovers will be performed on dca-acc-6k-2 while background test traffic and a measurable traffic stream are being run.

Test Procedure

The procedure used to perform the Access Layer Supervisor Failover Using SSO with NSF (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that both supervisors in dca-acc-6k-2 are online using the **show module** command.

The supervisors are listed in slots 5 and 6. One is listed as "Active" and the other as "Hot".

Step 3 Verify that the system is ready for an SSO failover using the **show redundancy states** command.

The operational redundancy mode is "sso" and the peer state is "STANDBY HOT".

Step 4 Begin running about an hour’s worth of traffic. This will include both the background test traffic and the measurable traffic stream.

Step 5 Once traffic is running, force a failover of the active supervisor on dca-acc-6k-2 using the **redundancy force-switchover** command.

Step 6 When the failed supervisor reboots and comes back online, verify that it is online using the **show module** command and that it is ready for another SSO redundancy failover using the **show redundancy states** command.

Step 7 Use the **show logging** command to verify that no errors occurred during the failover and recovery.

Step 8 Repeat the failover scenario 9 more times (for a total of ten). After each failover, verify the system status using the **show module**, **show redundancy states** and **show logging** commands.

Step 9 When all of the failovers are complete, analyze the traffic statistics for excessive errors and to verify that none of the failovers resulted in more than 3 seconds of traffic loss.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the system will recover from each forced failover successfully.
- We expect that the failed supervisor will come back online in the SSO standby role following each failover.
- We expect that each failover will result in less than 3 seconds of traffic loss.
- We expect no CPU or memory problems.

Results

Access Layer Supervisor Failover Using SSO with NSF (CSM Setup) passed.

Failure of Etherchannel Module on dca-agg-1 (CSM Setup)

The port-channel between the two Aggregation Layer device, dca-agg-1 and dca-agg-2, is critical for the monitoring of health and the synchronization of connections between the service modules. This link, an 802.1q trunk, carries the VLAN's that communicate the heartbeat messages between the CSM's and the FWSM's. Further, it replicates the connection states between the peer service modules so that downtime due to failover is minimized. The redundancy of this port-channel is essential.

In the DCAP test topology, this port-channel comprises two TenGigabitEthernet links, each link on a separate WS-X6704-10GE module. This test verified that if one of these modules were to reset, the impact to traffic would be minimal. Each of the two modules will be flapped multiple times. Client-to-server TCP traffic will be monitored to verify that there are no adverse effects.

Test Procedure

The procedure used to perform the Failure of Etherchannel Module on dca-agg-1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 4 Use the **show module** command to verify that the WS-X6704-10GE modules in slots 9 and 10 are online.

Step 5 On dca-agg-1, reset the TenGigabitEthernet module in slot 9 using the **hw-module module 9 reset** command.

Step 6 When module 9 comes back online, verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the **show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 7 Repeat the reset of module 9 on dca-agg-1 nine times for a total of ten flaps.

Step 8 Measure any traffic loss due to the module being reset.

Step 9 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 still consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 10 Use the **show module** command to verify that the WS-X6704-10GE modules in slots 9 and 10 are online.

Step 11 On dca-agg-1, reset the TenGigabitEthernet module in slot 10 using the **hw-module module 10 reset** command.

Step 12 When module 10 comes back online, verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 13 Repeat the reset of module 10 on dca-agg-1 nine times for a total of ten flaps.

Step 14 Measure any traffic loss due to the module being reset.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the port-channel interface to maintain normal operation as a logical interface when one of the hardware modules is reloaded.

- We expect any traffic loss due to the module reload to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of Etherchannel Module on dca-agg-1 (CSM Setup) passed.

Failure of Etherchannel Module on dca-agg-2 (CSM Setup)

The port-channel between the two Aggregation Layer device, dca-agg-1 and dca-agg-2, is critical for the monitoring of health and the synchronization of connections between the service modules. This link, an 802.1q trunk, carries the VLAN's that communicate the heartbeat messages between the CSM's and the FWSM's. Further, it replicates the connection states between the peer service modules so that downtime due to failover is minimized. The redundancy of this port-channel is essential.

In the DCAP test topology, this port-channel is composed of two TenGigabitEthernet links, each link on a separate WS-X6704-10GE module. This test verified that if one of these modules were to reset, the impact to traffic would be minimal. Each of the two modules will be flapped multiple times. Client-to-server TCP traffic will be monitored to verify that there are no adverse effects.

Test Procedure

The procedure used to perform the Failure of Etherchannel Module on dca-agg-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 4 Use the **show module** command to verify that the WS-X6704-10GE modules in slots 9 and 10 are online.

Step 5 On dca-agg-2, reset the TenGigabitEthernet module in slot 9 using the **hw-module module 9 reset** command.

Step 6 When module 9 comes back online, verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 7 Repeat the reset of module 9 on dca-agg-2 nine times for a total of ten flaps.

Step 8 Measure any traffic loss due to the module being reset.

Step 9 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 still consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 10 Use the **show module** command to verify that the WS-X6704-10GE modules in slots 9 and 10 are online.

Step 11 On dca-agg-2, reset the TenGigabitEthernet module in slot 10 using the **hw-module module 10 reset** command.

Step 12 When module 10 comes back online, verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 13 Repeat the reset of module 10 on dca-agg-2 nine times for a total of ten flaps.

Step 14 Measure any traffic loss due to the module being reset.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the port-channel interface to maintain normal operation as a logical interface when one of the hardware modules is reloaded.
- We expect any traffic loss due to the module reload to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of Etherchannel Module on dca-agg-2 (CSM Setup) passed with exception. The following exceptions were noted: CSCsk60108.

HSRP Failover with Fast Timers

Hot Standby Router Protocol (HSRP) is used to provide gateway redundancy to hosts on a given network. The Aggregation Layer devices dca-agg-1 and dca-agg-2 are configured to share a virtual IP address (VIP) that serves as the gateway for the hosts on that network. The router with the highest priority becomes the active HSRP router for that network, while the other becomes the standby. The two communicate through Hello packets. Should the standby router stop receiving Hello packets from the active for a period of time (called the Dead Time), it will assume the active is no longer available and transition itself from standby state to active state. This provides a high-availability gateway to the hosts on the network.

In the DCAP test topology, 201 VLAN's are configured with HSRP. The device dca-agg-1 is the active HSRP router, with a priority of 120 on each VLAN interface. The device dca-agg-2 is the standby HSRP router, with a priority of 110 on each VLAN interface. Each VLAN interface is configured with a separate HSRP group. This allows for each interface to have a unique MAC address. These two devices multicast their status with Hello packets sent every one second. The configured Dead Timer for each VLAN is three seconds.

Should dca-agg-1 fail, dca-agg-2 will assume the active HSRP router role after three seconds. If dca-agg-1 fails, dca-agg-2 will also assume the active role for other services, such as those provided by the CSM, FWSM, and SSLSM modules. Traffic will fail over completely to dca-agg-2 and its services modules.

When dca-agg-1 comes back online, it is intended that it will resume the active role for these various services, including HSRP. In order to avoid a flood of management traffic at the point when dca-agg-1 becomes available again, the HSRP configuration on the VLAN's specifies a wait period of 60 seconds. Even though dca-agg-2 is receiving Hello packets again from dca-agg-1, it will not give up the active role until it receives the Coup message that dca-agg-1 sends about 60 seconds after it comes back online.

This test verified that the HSRP protocol functions as configured in the DCAP test topology. The first part of this test proved the function on a small scale, one VLAN interface on dca-agg-1 was shut down. It was verified that dca-agg-2 took over the active role approximately three seconds after the VLAN on dca-agg-1 was shut down. The VLAN interface on dca-agg-1 was then brought back online, and it was verified that a Coup message was not sent until about 60 seconds after dca-agg-1 begins sending Hellos again.

The second part of this test verified that HSRP with these fast timers functioned correctly on a large scale, but shutting down 200 VLAN interfaces at once on dca-agg-1. The state transitions were monitored on dca-agg-2 to verify that all VLAN's transitioned after about three seconds. When the VLAN's on dca-agg-1 were brought back online, it was verified that, 60 seconds later dca-agg-1 became the active HSRP router again and dca-agg-2 transitioned back to standby for all of the VLAN's.

Note that traffic was not used in this test case as it looks more at the functionality of the protocol. The impacts on traffic were looked at during the *Reset of Aggregation Layer Device dca-agg-1* test case.

Test Procedure

The procedure used to perform the HSRP Failover with Fast Timers test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the HSRP configuration of the VLAN 1102 interface on dca-agg-1 using the **show running-config interface Vlan1102** command.

VLAN 1102 is configured with a real IP address of 101.1.2.2/24 and a virtual IP address of 101.1.2.1/24. The standby group for this VLAN is two, so the virtual MAC address for this gateway will be 0000.0c07.ac02. The priority is configured as 120 and the timers are configured as one second for the Hellos and three seconds for the Dead Timer. Preempt is configured, with a delay of 60 seconds.

Step 3 Verify the HSRP configuration of the VLAN 1102 interface on dca-agg-2 using the **show running-config interface Vlan1102** command.

VLAN 1102 is configured with a real IP address of 101.1.2.3/24 and a virtual IP address of 101.1.2.1/24. The standby group for this VLAN is 2, so the virtual MAC address for this gateway will be 0000.0c07.ac02. The priority is configured as 110 and the timers are configured as one second for the Hellos and three seconds for the Dead Timer. Preempt is configured, with a delay of 60 seconds.

Step 4 Verify that dca-agg-1 is the active HSRP router by using the **show standby vlan 1102** command on both dca-agg-1 and dca-agg-2.

On dca-agg-1, the output will display "Local state is Active" and a virtual MAC address of 0000.0c07.ac02. The output will also show that the router with address 101.1.2.3 is the standby HSRP router.

On dca-agg-2, the output will show "Local state is Standby" and no virtual MAC will be displayed. The output will also show that the router with address 101.1.2.2 is the active HSRP router.

Step 5 On dca-agg-2, set the **debug condition interface Vlan1102** so that only activity on this VLAN will be logged in the debugging that is about to take place.

Step 6 On dca-agg-2, turn on debugging for HSRP Hello and Coup packets using the **debug standby packets hello** and **debug standby packets coup** commands.

Step 7 While the debugs are active on dca-agg-2, shut down the VLAN 1102 interface on dca-agg-1.

Step 8 Using the debugs, verify that, on dca-agg-2, VLAN 1102 moves from standby to active state in two to three seconds.

In steady-state operation, there is a one for one exchange of Hello PDU's. When VLAN 1102 is shut down on dca-agg-1, dca-agg-2 will stop receiving Hellos. There should be, therefore, a period of two to three unilateral Hellos shown on dca-agg-2 before the state transition occurs.

Step 9 Verify that dca-agg-2 is now the active HSRP router by issuing the **show standby vlan 1102** command on both dca-agg-1 and dca-agg-2.

On dca-agg-1, the output will read "Local state is Init (interface down)".

On dca-agg-2, the output will read "Local state is Active" and, now, a virtual MAC of 0000.0c07.ac02 will be displayed.

Step 10 Bring the VLAN 1102 interface back online on dca-agg-1 using the **no shutdown** command.

Step 11 Verify that dca-agg-2 starts receiving Hello packets from dca-agg-1 again and that, after about 60 seconds, a Coup message is received and a state transition occurs.

About a minute after dca-agg-2 begins receiving Hellos again from dca-agg-1, it will receive a Coup message. When this Coup message is received, dca-agg-2 will transition from standby to speak state.

Step 12 Verify that dca-agg-1 is the active HSRP router by using the **show standby vlan 1102** command on both dca-agg-1 and dca-agg-2.

On dca-agg-1, the output will show "Local state is Active" as well as a virtual MAC address of 0000.0c07.ac02. The output will show that the router with address 101.1.2.3 is the standby HSRP router.

On dca-agg-2, the output will show "Local state is Standby" and no virtual MAC will be displayed. The output will also show that the router with address 101.1.2.2 is the active HSRP router.

Step 13 On dca-agg-2, turn off all debugging and unset the debug condition using the **undebug all** and **no debug condition all** commands.

Step 14 Use the **show standby brief** command on dca-agg-1 and dca-agg-2 to verify that dca-agg-1 is the active HSRP router for all VLAN's.

On dca-agg-1, under the state column heading, each VLAN should read "Active". On dca-agg-2, each VLAN should read "Standby".

Step 15 Use the **interface range** and **shutdown** commands to shut down VLAN's 1101 to 1300 on dca-agg-1.

Step 16 By watching the error log messages, verify visually that the VLAN's transition to init state on dca-agg-1 and to active state on dca-agg-2 within the three second Dead Timer window.

Step 17 Verify that the 200 VLAN's are now in init state on dca-agg-1 and active state on dca-agg-2 using the **show standby brief** command.

Step 18 Use the **interface range** and **no shutdown** commands to bring online VLAN's 1101-1300 on dca-agg-1.

Step 19 By watching the error log messages, verify visually that the VLAN's transition to active state on dca-agg-1 and to standby state on dca-agg-2 after about 60 seconds.

Step 20 Verify that the 200 VLAN's are again in active state on dca-agg-1 and standby state on dca-agg-2 using the **show standby brief** command.

Step 21 Stop background scripts to collect final status of network devices and analyze for error.

Step 22 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the standby HSRP router to become active following a loss of 2 to 3 hello PDU's.
- We expect proper failover to occur on both a small scale and a large scale.
- We expect the HSRP router with preempt configured to resume the active role after the configured wait time.
- We expect preemption to work correctly on both a small scale and a large scale.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

HSRP Failover with Fast Timers passed.

HSRP Recovery From System Failure

The device dca-agg-1 is, by configuration, the active HSRP router for 201 VLAN's in the DCAP test topology. It is also configured with HSRP preempt, and a preempt delay of 60 seconds. This means that should a failover occur, and dca-agg-2 becomes the active router, when dca-agg-1 comes back online it will restore active state 60 seconds after it becomes available.

This test verified the behavior of HSRP during a system failure. The first part of the test defined what occurred when dca-agg-1 was rebooted. By design, dca-agg-2 became the new active HSRP router once dca-agg-1 went offline. This happened within three seconds (the configured Dead Timer) of dca-agg-1 leaving the topology.

The second part of this test verified that dca-agg-1, once it became available again, preempted for active HSRP status after 60 seconds of participating in the Hello exchange with dca-agg-2.

No traffic was used during this test because it was focused on looking at the HSRP behavior aspect of a failover scenario. The test *Reset of Aggregation Layer Device dca-agg-1* defined the effects of a fail over on traffic.

Test Procedure

The procedure used to perform the HSRP Recovery From System Failure test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that dca-agg-1 is the active HSRP router for VLAN's 301 and 1101 to 1300 using the **show standby brief** command.

In the output, under the column headed "State", each VLAN should be listed as "Active".

Step 3 Verify that dca-agg-2 is the standby HSRP router for VLAN's 301 and 1101 to 1300 using the **show standby brief** command.

In the output, under the column headed "State", each VLAN should be listed as "Standby".

Step 4 Reload dca-agg-1.

Step 5 Verify that the VLAN's in dca-agg-2 transitioned from standby to active state within two to three seconds following the reload using the **show standby brief** command.

Step 6 Wait for dca-agg-1 to come back online.

Step 7 When dca-agg-1 comes back online, verify that it preempts dca-agg-2 and resumes the active HSRP role after about 60 seconds of being online. Use the **show standby brief** command on both dca-agg-1 and dca-agg-2 for this.

Device dca-agg-1 should read "Active" for all VLAN's again, and dca-agg-2 should read "Standby".

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect HSRP to fail over correctly when the active HSRP router is taken out of service.
- We expect the router to be brought back online and resume its active HSRP role after the configured 60-second delay.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

HSRP Recovery From System Failure passed.

Repeated Reset of Standby Supervisor in Access Layer (CSM Setup)

The Catalyst 6500 systems used in the Access Layer in the DCAP test topology are equipped with dual supervisors for intra-chassis redundancy. The resetting of the standby supervisor in either of these systems should not result in any errors in the system or impact to traffic. This test verified that neither resulted from a repeated reset of the standby supervisor in dca-acc-6k-2.

Test Procedure

The procedure used to perform the Repeated Reset of Standby Supervisor in Access Layer (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that both supervisors in dca-acc-6k-2 are online using the **show module** command.

The supervisors are listed in slots 5 and 6. One is listed as "Active" and the other as "Hot".

Step 3 Begin running about an hour's worth of traffic. This will include both the background test traffic and the measurable traffic stream.

Step 4 Once traffic is running, reset the standby supervisor on dca-acc-6k-2 (the one shown as "Hot" in the **show module** output) using the **hw-module module***module***reset** command.

Step 5 When the standby supervisor reboots and comes back online, verify that it is online using the **show module** and again in the "Hot" standby state.

Step 6 Repeat the standby supervisor reset scenario 9 more times (for a total of ten). After each reset, verify the system status using the **show module** command.

Step 7 When all of the failovers are complete, analyze the traffic statistics for excessive errors and to verify that none of the failovers resulted in more than 3 seconds of traffic loss.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the resetting of the standby supervisor in the Access Layer device will not cause any ill-effects to the system or to traffic in the DCAP test topology.
- We expect no CPU or memory problems.

Results

Repeated Reset of Standby Supervisor in Access Layer (CSM Setup) passed.

Reset of Aggregation Layer Device dca-agg-1 (CSM Setup)

The Aggregation Layer provides the bulk of services for traffic coming into the data center. Services such as server load balancing, SSL decryption and encryption, and firewalling are provided by the Aggregation Layer devices dca-agg-1 and dca-agg-2.

Redundancy in the aggregation layer is important for providing a high level of availability for these services. The DCAP test topology was designed to provide redundancy through a pair of Aggregation Layer devices rather than redundant supervisors because the failover timers for many of the services are set low. This means that a service failover could be triggered even by a fast SSO/NSF failover. This is inline with the goals of predictability with regards to traffic in the data center. If anything less than all services fail over, the result is an unclean picture of traffic using both Aggregation Layer boxes for the various services before arriving at the destination.

By configuration, dca-agg-1 is the primary provider of services in the DCAP test topology. It is the STP root, the Active HSRP router, and it houses the active CSM and FWSM. The standby services wait on dca-agg-2 for a failover event.

This test verified the impact on traffic due to the failure of the Aggregation device dca-agg-1. Background traffic was run using Linux servers and measurable traffic using the Shenick test tool. A total of five system resets were performed.

Test Procedure

The procedure used to perform the Reset of Aggregation Layer Device dca-agg-1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Start the background traffic and the measurable Shenick traffic.

Step 3 Once traffic is running, verify that dca-agg-1 is passing traffic through interfaces Te9/1, Te9/2, Te9/4, Te10/1, Te10/2, and Te10/4 using the **show interfaces counters | include Port|Te9|Te10** command.

Step 4 Use the **show interfaces counters | include Port|Te9|Te10** command to verify that dca-agg-2 is not passing any substantial traffic.

There will be a variable amount of management traffic being passed by dca-agg-2.

Step 5 Reload dca-agg-1.

Step 6 Use the **show interfaces counters | include Port|Te9|Te10** command to verify that dca-agg-2 is now passing traffic.

Step 7 When dca-agg-1 comes back online, verify that it is passing traffic through interfaces Te9/1, Te9/2, and Po1 using the **show interfaces counters | include Port|Te9|Te10|Po1** command.

The command in this step asks to look at the traffic counters on Port-channel 1 as well. This is because, following the recovery of dca-agg-1, the FWSM will remain active in dca-agg-2. The version of FWSM code that is running in DCAP Phase One does not have a preempt feature, and so the FWSM in dca-agg-1 never resumes the active role. This is why below, a manual reset of the FWSM in dca-agg-2 is called for.

This is also the reason that no traffic is seen from dca-agg-1 down to the Access Layer switches through interfaces Te9/4, Te10/1, Te10/2, or Te10/4. In order for traffic to pass between client and server, it must go through the active FWSM, which is now in dca-agg-2 (and will remain so indefinitely).

Step 8 Use the **show interfaces counters | include Port|Te9|Te10|Po1** command to verify that dca-agg-2 is passing traffic over Port-channel 1 and the interfaces that connect to the downstream Access Layer devices, Te9/4, Te10/1, Te10/2, and Te10/4.

Step 9 Reboot the FWSM in dca-agg-2 using the **hw-module module 1 reset** command.

This will force the FWSM in dca-agg-1 back into active mode, the starting point for this failover test.

Step 10 Verify that dca-agg-1 is again passing traffic through interfaces Te9/4, Te10/1, Te10/2, and Te10/4 using the **show interfaces counters | include Port|Te9|Te10|Po1** command.

Step 11 Repeat the above sequence of reloading the DUT and checking counters four times.

Step 12 Verify the traffic that was lost via the Shenick test tool and the Linux-generated tool.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect there to be traffic loss on the order of five seconds or less for each failover performed.
- We expect the reset device comes back online ande resume forwarding traffic as before.

- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Reset of Aggregation Layer Device dca-agg-1 (CSM Setup) passed.

Reset of Aggregation Layer Device dca-agg-2 (CSM Setup)

The Aggregation Layer provides the bulk of services for traffic coming into the data center. Services such as server load balancing, SSL decryption and encryption, and firewalling are provided by the Aggregation Layer devices dca-agg-1 and dca-agg-2.

Redundancy in the aggregation layer is important for providing a high level of availability for these services. The DCAP test topology was designed to provide redundancy through a pair of Aggregation Layer devices rather than redundant supervisors because the failover timers for many of the services are set very low. This means that a service failover could be triggered even by a fast SSO/NSF failover. This is in-line with the goals of predictability with regards to traffic in the data center. If anything less than all services fail over, the result is a unclean picture of traffic using both Aggregation Layer boxes for the various services before arriving at the destination.

By configuration, dca-agg-2 is the standby provider of services in the DCAP test topology. It is the STP secondary root, the standby HSRP router, and it houses the standby CSM and FWSM. These standby services wait on dca-agg-2 for a failover event.

This test verified the impact on traffic due to the failure of the Aggregation device dca-agg-2. Background traffic was run using Linux servers and measurable traffic using the Shenick test tool. A total of five system resets were performed.

Test Procedure

The procedure used to perform the Reset of Aggregation Layer Device dca-agg-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Start the background traffic and the measurable Shenick traffic.

Step 3 Once traffic is running, verify that dca-agg-2 is not passing any significant data traffic through interfaces Te8/1, Te8/2, Te9/4, Te9/1, Te9/2, Po 1 using the **show interface | include packets/sec** command.

The four TenGigabit Ethernet interfaces are connected downstream to the Access Layer switches. Because dca-agg-2 is not providing any active services for data center traffic, other than SSL decryption/encryption, no data traffic should be transiting dca-agg-2. The EtherChannel Po1 connects dca-agg-2 to dca-agg-1 and, in steady-state, is used solely for management traffic.

Note that the output will show significant traffic being sent to the Access Layer devices. This is traffic from the Shenick test tool that is being flooded as a result of the tool not answering periodic ARP requests.

Step 4 Reload dca-agg-2.

Step 5 When dca-agg-2 comes back online, verify that it is, again, not passing any substantial traffic using the **show interfaces counters | include Port|Te9|Te10|Po1** command.

Note that in the output, it will show significant traffic being sent to the Access Layer devices. This is traffic from the Shenick test tool that is being flooded as a result of the tool not answering periodic ARP requests.

Step 6 Repeat the above sequence of reloading dca-agg-2 and checking counters four times.

Step 7 Verify the traffic that was lost via the Shenick test tool and the Linux-generated tool.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect there to traffic loss on the order of three seconds or less for each failover performed.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Reset of Aggregation Layer Device dca-agg-2 (CSM Setup) passed.

Reset of Core Layer Device dca-core-1 (CSM Setup)

The Core Layer is the first stop for traffic coming into the data center. As such, redundancy in the core layer is important for maintaining a high level of availability. Having redundant devices running OSPF allows for failover times nearly as fast as could be achieved through redundant supervisors running SSO with NSF. Further, redundant devices provides load-balancing mechanisms.

This test verified the impact on traffic due to the failure of the core device dca-core-1. Background traffic was run using Linux servers and measurable traffic using the Shenick test tool. Five system resets were performed.

Test Procedure

The procedure used to perform the Reset of Core Layer Device dca-core-1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Start the background traffic and the measurable Shenick traffic.

Step 3 Once traffic is running, verify that dca-core-1 is passing traffic through interfaces Te1/2 and Te1/3 using the **show interface | include packets/sec** command.

Step 4 Reload dca-core-1.

Step 5 When dca-core-1 comes back online, verify that it is again passing traffic through interfaces Te1/2 and Te1/3 using the **show interface | include packets/sec** command.

Step 6 Repeat the above sequence of reloading the DUT and checking counters four times.

Step 7 Verify the traffic that was lost via the Shenick test tool and the Linux-generated tool.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect there to be traffic loss on the order of three seconds or less for each failover performed.
- We expect the reset device to come back online and resume forwarding traffic as before.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Reset of Core Layer Device dca-core-1 (CSM Setup) passed.

Reset of Core Layer Device dca-core-2 (CSM Setup)

The Core Layer is the first stop for traffic coming into the data center. As such, redundancy in the core layer is important for maintaining a high level of availability. Having redundant devices running OSPF allows for failover times nearly as fast as could be achieved through redundant supervisors running SSO with NSF. Further, redundant devices provides load-balancing mechanisms.

This test verified the impact on traffic due to the failure of the core device dca-core-2. Background traffic was run using Linux servers as well as measurable traffic using the Shenick test tool. A total of five system resets were performed.

Test Procedure

The procedure used to perform the Reset of Core Layer Device dca-core-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Start the background traffic and the measurable Shenick traffic.

Step 3 Once traffic is running, verify that dca-core-2 is passing traffic through interfaces Te1/2 and Te1/3 using the **show interfaces | i packets/sec** command.

Step 4 Reload dca-core-2.

Step 5 When dca-core-2 comes back online, verify that it is again passing traffic through interfaces Te1/2 and Te1/3 using the **show interface | include packets/sec** command.

Step 6 Repeat the above sequence of reloading the DUT and checking counters four times.

Step 7 Verify the traffic that was lost via the Shenick test tool and the Linux-generated tool.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect there to be traffic loss for three seconds or less for each failover performed.
- We expect the reset device to come back online and resume forwarding traffic as before.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Reset of Core Layer Device dca-core-2 (CSM Setup) passed.

Spanning-Tree Primary Root Failure and Recovery (CSM Setup)

In the DCAP test topology, dca-agg-1 is configured as the primary spanning-tree root switch. This means that all traffic flowing between clients and servers will find its way through dca-agg-1.

The spanning-tree protocol has certain rules governing the STP link states (forwarding and blocking) of root and non root devices. This test verified that the interfaces in the Layer 2 domain of the DCAP test topology were in the proper STP state initially, following a primary root failure, and after the primary root recovery.

It should be noted that the purpose of this test is to look specifically at the functionality and behavior of the Rapid PVST+ spanning-tree protocol during the failure of the primary root switch. No traffic is run in this test for this reason. Other tests in this suite that measure the impact of the failure of certain devices on traffic.

Test Procedure

The procedure used to perform the Spanning-Tree Primary Root Failure and Recovery (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 In spanning-tree, the bridge with the lowest configured priority becomes the STP root switch. If all priorities are the same, that is default, the bridge with the lowest MAC address becomes root. Use the **show spanning-tree bridge** command to verify that dca-agg-1 is configured with the lowest STP priority for all VLAN's in the Layer 2 domain. Also verify that dca-agg-2 is configured with the second-lowest priority.

The configured STP priority of dca-agg-1 is 24576 for all VLAN's. Because **spanning-tree extend system-id** is configured, the real priority of dca-agg-1 for a particular VLAN will be the sum of the configured priority plus the value of that VLAN.

The configured STP priority of dca-agg-1 is 28672 for all VLAN's. The adjustment to real/advertised priority due to the **spanning-tree extend system-id** command applies to this switch as well. Note that because dca-agg-2 has the second-highest STP priority of all the switches in the L2 domain, it is next line to become STP root should dca-agg-1 fail.

The STP priority of the other switches in the L2 domain was left as default, with the advertised priority being the default value plus the VLAN value.

Step 3 Use the **show spanning-tree summary** command on dca-agg-1 and dca-agg-2 to verify that all VLAN's in each of these switches are in STP forwarding state.

Step 4 Use the **show spanning-tree summary** command on dca-acc-6k-1, dca-acc-6k-2, and dca-voodoo-2 to verify that one interface in each VLAN on these switches is in STP blocking state.

Each of these three switches shows 202 VLAN's. Each switch is dual-homed to the two Aggregation Layer switches, so that their Layer 2 design is a triangle-shaped looped topology. As such, one uplink interface will be forwarding for all VLAN's (the one connected to the STP Root) and one uplink will be blocking for all VLAN's (the one connected to the STP Secondary Root). So, 202 interfaces are in blocking state.o

Step 5 Use the **show spanning-tree summary** command on dca-acc-4k-1 and dca-acc-4k-2 to verify the STP states of the VLAN's in these switches.

Both of these switches show 202 VLAN's. These two switches are connected to the Aggregation Layer switches and each other, such that their Layer 2 design is a U-shaped looped topology. As such, all the interfaces on one of the two (dca-acc-4k-1) will be forwarding for all VLAN's, and one switch (dca-acc-4k-2) will have one interface in blocking state for all VLAN's. So, 202 interfaces are in blocking state.

Step 6 Use the **show spanning-tree vlan 2101** command on all seven Layer 2 devices to verify their steady-state status. This VLAN will be looked at as a representative of the whole set when the failover, and recovery is performed in later steps.

All seven devices show that the MAC address of the Root switch is 00d0.04ac.f400 (dca-agg-1).

The device dca-agg-1 should show six interfaces in STP FWD state. Te9/4, Te10/1, Te10/2, and Te10/4 are trunks to the Access Layer devices. Po1 is the trunk connecting dca-agg-1 to dca-agg-2, used to carry the fault-tolerant and synchronization information for the CSM and FWSM. Po270 is the internal EtherChannel interface by which the FWSM connects to the system backplane.

The device dca-agg-2 should also show six interfaces in STP FWD state. Te9/4, Te10/1, Te10/2, and Te10/4 are trunks to the Access Layer devices. Po1 is the trunk connecting dca-agg-2 to dca-agg-1, which is used to carry the fault-tolerant and synchronization information for the CSM and FWSM. Po270 is the internal EtherChannel interface by which the FWSM connects to the system backplane.

The device dca-acc-6k-1 should show several interfaces in STP FWD state and one in BLK state. Te1/1 is the Root Port for this device, connecting it to the STP root (dca-agg-1), and is forwarding. Te1/2 is the Alternate Port, connecting dca-acc-6k-1 to the secondary root (dca-agg-2), and is blocking. The remainder of the ports are connected to servers, and are all in forwarding state.

The device dca-acc-6k-2 should show several interfaces in STP FWD state and one in BLK state. Te1/1 is the Root Port for this device, connecting it to the STP root (dca-agg-1), and is forwarding. Te1/2 is the Alternate Port, connecting dca-acc-6k-2 to the secondary root (dca-agg-2), and is blocking. The remainder of the ports are connected to servers, and are all in forwarding state.

The device dca-acc-4k-1 should show all interfaces in STP FWD state. Te1/49 is the Root Port for this device, connecting it to the STP root (dca-agg-1), and is forwarding. Te1/50 connects dca-acc-4k-1 to the other Cat4k access switch (dca-acc-4k-2), and is also forwarding. The remainder of the ports are connected to servers, and are all in forwarding state.

The device dca-acc-4k-2 should show only one interface in STP FWD state and one in STP BLK state. Te1/49 is the Root Port for this device, connecting it to the secondary STP root (dca-agg-2), and is forwarding. Te1/50 connects dca-acc-4k-1 to the other Cat4k access switch (dca-acc-4k-1), and is blocking. There are no server ports in VLAN 2101 on this access switch.

The device dca-voodoo-2 should show two interfaces in STP FWD state and one in BLK state. Te1/1 is the Root Port for this device, connecting it to the STP root (dca-agg-1), and is forwarding. Te1/2 connects dca-voodoo-2 to the secondary root switch (dca-agg-2), and is blocking. The only other interface (Gi6/1) in the list is a server port and is forwarding.

Step 7 Use the **reload** command on the primary STP root, dca-agg-1.

Step 8 About five seconds after reloading dca-agg-1, check the STP states for VLAN 2101 again, on the six remaining Access Layer switches using the **show spanning-tree vlan 2101** command. Verify that there are no ports in the blocking state anymore, and that all devices show the MAC address of the Root switch as being 0007.ec73.d000 (dca-agg-2).

The spanning-tree protocol that is being used is Rapid PVST+, or rPVST+. It facilitates sub second state change times. Normal spanning-tree (IEEE 802.1d) takes roughly 30 seconds to reconverge following a topology change.

Step 9 Use the **show spanning-tree summary** command on all Access Layer devices to verify that there are no VLAN's in blocking state.

Step 10 Wait for the original STP Root device, dca-agg-1, to come back online.

Step 11 Once dca-agg-1 is online again and operational, issue the **show spanning-tree vlan 2101** command on all seven Layer 2 devices to verify their status has returned to steady-state conditions.

All seven devices again show that the MAC address of the Root switch is **00d0.04ac.f400** (dca-agg-1).

The device dca-agg-1 should show six interfaces in STP FWD state. Te9/4, Te10/1, Te10/2 and Te10/4 are trunks to the Access Layer devices. Po1 is the trunk connecting dca-agg-1 to dca-agg-2, used to carry the fault-tolerant and synchronization information for the CSM and FWSM. Po270 is the internal EtherChannel interface by which the FWSM connects to the system backplane.

The device dca-agg-2 should show six interfaces in STP FWD state. Te9/4, Te10/1, Te10/2 and Te10/4 are trunks to the Access Layer devices. Po1 is the trunk connecting dca-agg-2 to dca-agg-1, used to carry the fault-tolerant and synchronization information for the CSM and FWSM. Po270 is the internal EtherChannel interface by which the FWSM connects to the system backplane.

The device dca-acc-6k-1 should show several interfaces in STP FWD state and one in BLK state. Te1/1 is the Root Port for this device, connecting it to the STP root (dca-agg-1), and is forwarding. Te10/2 is the Alternate Port, connecting dca-acc-6k-1 to the secondary root (dca-agg-2), and is blocking. The remainder of the ports are connected to servers, and are all in forwarding state.

The device dca-acc-6k-2 should show several interfaces in STP FWD state and one in BLK state. Te1/1 is the Root Port for this device, connecting it to the STP root (dca-agg-1), and is forwarding. Te10/2 is the Alternate Port, connecting dca-acc-6k-2 to the secondary root (dca-agg-2), and is blocking. The remainder of the ports are connected to servers, and are all in forwarding state.

The device dca-acc-4k-1 should show all interfaces in STP FWD state. Te1/49 is the Root Port for this device, connecting it to the STP root (dca-agg-1), and is forwarding. Te1/50 connects dca-acc-4k-1 to the other Cat4k access switch (dca-acc-4k-2), and is also forwarding. The remainder of the ports are connected to servers, and are all in forwarding state.

The device dca-acc-4k-2 should show only one interface in STP FWD state and one in STP BLK state. Te1/49 is the Root Port for this device, connecting it to the secondary STP root (dca-agg-2), and is forwarding. Te1/50 connects dca-acc-4k-1 to the other Cat4k access switch (dca-acc-4k-1), and is blocking. There are no server ports in VLAN 2101 on this Access switch.

The device dca-voodoo-2 should show two interfaces in STP FWD state and one in BLK state. Te1/1 is the Root Port for this device, connecting it to the STP root (dca-agg-1), and is forwarding. Te1/2 connects dca-voodoo-2 to the secondary root switch (dca-agg-2), and is blocking. The only other interface (Gi6/1) in the list is a server port and is forwarding.

Step 12 Use the **show spanning-tree summary** command on dca-agg-1 and dca-agg-2 to verify that all VLAN's in each of these switches are in STP forwarding state.

Step 13 Use the **show spanning-tree summary** command on dca-acc-6k-1, dca-acc-6k-2, and dca-voodoo-2 to verify that there is one interface in each VLAN on these switches that is in STP blocking state.

Each of these three switches shows 202 VLAN's. Each switch is dual-homed to the two Aggregation Layer switches, so that their Layer 2 design is a triangle-shaped looped topology. As such, one uplink interface will be forwarding for all VLAN's (the one connected to the STP Root) and one uplink will be blocking for all VLAN's (the one connected to the STP Secondary Root). So, 202 interfaces are in blocking state.

Step 14 Use the **show spanning-tree summary** command on dca-acc-4k-1 and dca-acc-4k-2 to verify the STP states of the VLAN's in these switches.

Both of these switches show 202 VLAN's. These two switches are connected to the Aggregation Layer switches and each other, such that their Layer 2 design is a U-shaped looped topology. As such, all the interfaces on one of the two (dca-acc-4k-1) will be forwarding for all VLAN's and one switch (dca-acc-4k-2) will have one interface in blocking state for all VLAN's. So, 202 interfaces are in blocking state.

Step 15 Repeat the above sequence four times, gathering the information necessary to verify correct STP behavior each time.

Step 16 Stop background scripts to collect final status of network devices and analyze for error.

Step 17 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect spanning-tree to reconverge appropriately when the primary root switch is taken offline.
- We expect spanning-tree to reconverge appropriately when the primary root switch is recovered.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Spanning-Tree Primary Root Failure and Recovery (CSM Setup) passed.

Link Failure

The tests included in this section measure the impact of a link failure occurring in the data path. The ability of the data center infrastructure to respond favorable to such scenarios is critical to the high availability of network resources.

This section contains the following topics:

Failure of 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Access Layer device, dca-acc-4k-1, and an Access Layer device, dca-acc-4k-2. Web traffic was sent using a test tool to a VIP on the CSM. Te1/50 on dca-acc-4k-1 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-acc-4k-1 and shut down interface Te1/50.

Step 4 After two minutes, bring interface Te1/50 back online using the no shutdown command on dca-acc-4k-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2 (CSM Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and an Access Layer device, dca-acc-4k-1. Web traffic was sent using a test tool to a VIP on the CSM. Te9/6 on dca-agg-1 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-1 and shut down interface Te9/6.

Step 4 After two minutes, bring interface Te9/6 back online using the no shutdown command on dca-agg-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1 (CSM Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and an Access Layer device, dca-acc-6k-1. Web traffic was sent using test tool to a VIP on the CSM. Te9/4 on dca-agg-1 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-1 and shut down interface Te9/4.

Step 4 After two minutes, bring interface Te9/4 back online using the no shutdown command on dca-agg-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1 (CSM Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and an Access Layer device, dca-acc-6k-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te10/1 on dca-agg-1 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-1 and shut down interface Te10/1.

Step 4 After two minutes, bring interface Te10/1 back online using the **no shutdown** command on dca-agg-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2 (CSM Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and an Access Layer device, dca-acc-4k-2. Web traffic was sent using a test tool to a VIP on the CSM. Te8/2 on dca-agg-2 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-2 and shut down interface Te8/2.

Step 4 After two minutes, bring interface Te8/2 back online using the no shutdown command on dca-agg-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.

- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2 (CSM Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and an Access Layer device, dca-acc-6k-1. Web traffic was sent using a test tool to a VIP on the CSM. Te9/4 on dca-agg-2 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-2 and shut down interface Te9/4.

Step 4 After two minutes, bring interface Te9/4 back online using the no shutdown command on dca-agg-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1 (CSM Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and an Access Layer device, dca-acc-6k-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te10/1 on dca-agg-2 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-2 and shut down interface Te10/1.

Step 4 After a minute, bring interface Te10/1 back online using the **no shutdown** command on dca-agg-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2 (CSM Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1 (CSM Setup)

This test measured the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and a Core Layer device, dca-core-1. Web traffic is sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/2 on dca-core-1 is shut down for two minutes then brought back online for two minutes. This is repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-1 and shut down interface Te1/2.

Step 4 After a minute, bring interface Te1/2 back online using the **no shutdown** command on dca-core-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1 (CSM Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and a Core Layer device, dca-core-1. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/3 on dca-core-1 was shut down for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-1 and shut down interface Te1/3.

Step 4 After a minute, bring interface Te1/3 back online using the **no shutdown** command on dca-core-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2 (CSM Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between the two Core Layer devices, dca-core-1 and dca-core-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/1 on dca-core-1 was shut down for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-1 and shut down interface Te1/1.

Step 4 After a minute, bring interface Te1/1 back online using the **no shutdown** command on dca-core-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.

- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2 (CSM Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and a Core Layer device, dca-core-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/2 on dca-core-2 was shut down for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-2 and shut down interface Te1/2.

Step 4 After a minute, bring interface Te1/2 back online using the **no shutdown** command on dca-core-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1 (CSM Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2 (CSM Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and a Core Layer device, dca-core-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/3 on dca-core-2 was shut down for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-2 and shut down interface Te1/3.

Step 4 After a minute, bring interface Te1/3 back online using the **no shutdown** command on dca-core-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2 (CSM Setup) passed.

Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2 (CSM Setup)

The port-channel between the two Aggregation Layer device, dca-agg-1 and dca-agg-2, is critical for the monitoring of health and the synchronization of connections between the service modules. This link, an 802.1q trunk, carries the VLAN's that communicate the heartbeat messages between the CSM's and the FWSM's. Further, it replicates the connection states between the peer service modules so that downtime due to failover is minimized. The redundancy of this port-channel is essential.

This test verified that if a single link of that port-channel were to go down, the impact to traffic would be minimal. Each of the two TenGigabitEthernet links in the port-channel were flapped multiple times. Client-to-server TCP traffic was monitored to verify that there were no adverse effects.

Test Procedure

The procedure used to perform the Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2 (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 4 On dca-agg-1, shut down interface Te9/3.

Step 5 After a minute, bring interface Te9/3 back online using the **no shutdown** command on dca-agg-1.

Step 6 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 7 Repeat the flap of interface Te9/3 on dca-agg-1 nine times for a total of ten flaps.

Step 8 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 9 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 still consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 10 On dca-agg-1, shut down interface Te10/3.

Step 11 After a minute, bring interface Te10/3 back online using the **no shutdown** command on dca-agg-1.

Step 12 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 13 Repeat the flap of interface Te10/3 on dca-agg-1 nine times for a total of ten flaps.

Step 14 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the port-channel interface to maintain normal operation as a logical interface when a single bundled link is flapped.
- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2 (CSM Setup) passed.

Network Resiliency Test (CSM Setup)

This is a single test that encompasses a suite of scripts used to take an initial snapshot of the test network, introduce mayhem, and take a final snapshot of the test network. The initial and final snapshots should be nearly identical, save for some cosmetic differences.

The class of test scripts used are called spiders, or crawlers. Each script was given a seed device to start at. It took a snapshot of the information for that crawler, then discovered the neighboring devices, and moved on to those neighboring devices to take snapshots of them, and so forth until all the devices in the network were covered. Nine individual crawler scripts were run at the beginning of this test. They reviewed the module status in the devices, the interface status, the trunk and channel status, and the status of certain protocols (CDP, UDLD, PIM, HSRP, and OSPF). Information was gathered from the device, and saved in a file, during this initial run.

After the initial snapshot is taken, a script called Rolling Havoc is run. This crawler logs into each network device and flaps each inter switch link a configured number of times.

After Rolling Havoc wrought its harm on the network, the nine other crawler scripts were run again, gathering post havoc information about the same aspects of the network. Because no other negative tests were taking place during this test sequence, the network returned to the identical state to what it was in before the Rolling Havoc script was run.

Test Procedure

The procedure used to perform the Network Resiliency Test (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Baseline all CDP neighbor relationships. Run the CDP crawler script verifying all expected CDP neighbors are reported.

The purpose of the CDP crawler script is to crawl through the network continuously, noting any changes that occur between traversals in CDP information. It parses information gathered from select CDP and Cisco IOS commands.

Step 3 Baseline all EtherChannel members. Run the channel crawler script verifying that all interfaces expected to be in channels are reported.

The purpose of the channel crawler script is to run through a network and verify that EtherChannels are in a proper state. It parses information gathered from select EtherChannel and Cisco IOS commands.

Step 4 Baseline all trunk interfaces. Run the trunk crawler script verifying that all expected trunking interfaces, configuration, and status are reported.

The purpose of the trunk crawler script is to run through a network and verify that trunking is in a proper state. It parses information gathered from select trunking and Cisco IOS commands.

Step 5 Baseline all interface states and counters. Run the interface crawler script recording interface counters and states.

The interface crawler script crawls through a network continually. All up/up interfaces are checked for various errors. Initially all non zero error counters will be logged, then any counters that increment from that point on.

Step 6 Baseline all interface UDLD states. Run the UDLD crawler script recording the UDLD state of all interfaces.

The UDLD crawler script gathers a list of UDLD ports from a list of devices and traverses their neighbors continuously checking for UDLD problems or inconsistencies. It parses information gathered from select UDLD and Cisco IOS commands.

Step 7 Baseline all linecards used in the topology. Run the module crawler script recording module counters and state.

The module crawler script gathers a list of modules from a list of devices and looks for problems or inconsistencies. It parses information gathered from select module and Cisco IOS commands.

Step 8 Baseline the HSRP feature in the topology. Run the HSRP crawler script recording HSRP state.

Step 9 Flap each of the active non management interfaces in the SH3 network five times each.

Step 10 Execute the CDP crawler script to verify that the CDP feature is still operating correctly in the Data Center test network.

Step 11 Execute the channel crawler script to verify that the EtherChannel feature is still operating correctly in the Data Center test network.

Step 12 Execute the trunk crawler script to verify that the trunking feature is still operating correctly in the Data Center test network.

Step 13 Execute the interface crawler script to verify that the basic functionality of the interface is still operating correctly in the Data Center test network.

Step 14 Execute the UDLD crawler script to verify that the UDLD feature is still operating correctly in the Data Center test network.

Step 15 Execute the module crawler script to verify that the line cards in the Data Center test network are still operating correctly.

Step 16 Execute the HSRP crawler script to verify that HSRP in the Data Center test network is still operating correctly.

Step 17 Stop background scripts to collect final status of network devices and analyze for error.

Step 18 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the EtherChannel feature to work correctly before and after the interface flapping.

- We expect the HSRP feature to work correctly before and after the interface flapping.
- We expect the CDP feature to work correctly before and after the interface flapping.
- We expect the trunking feature to work correctly before and after the interface flapping.
- We expect the PIM feature to work correctly before and after the interface flapping.
- We expect the UDLD feature to work correctly before and after the interface flapping.
- We expect the modules to work correctly before and after the interface flapping.
- We expect that there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Network Resiliency Test (CSM Setup) passed.

CHAPTER 3

Layer 2-3 Infrastructure with ACE

Layer 2-3 ACE testing reflects a design in which the ACE was used in the Aggregation Layer switches. The testing in this chapter is a subset of testing that was performed in Layer 2-3 Infrastructure with CSM, page 2-1, against the CSM/SSLSM combination. The tests that were repeated were those in which the traffic path had changed with the introduction of the ACE module as a substitute for the CSM/SSLSM. This included those tests that used traffic traversing the Aggregation Layer switches. Other tests that have been repeated include those in which the basic system functionality was changed due to the use of the ACE module.

Test Results Summary

Table 3-1 on page 3-1 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 3-1 on page 3-1 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 3-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Baseline, page 3-3	Baseline, page 3-3	1. Topology Baseline (ACE Setup)	
Baseline, page 3-3	Device Management, page 3-5	1. General On-Line Diagnostics (GOLD) (ACE Setup)	
Baseline, page 3-3	Traffic Forwarding, page 3-7	1. Zero Packet Loss (ACE Setup)	
Layer 3 Protocols, page 3-8	Hot Standby Router Protocol (HSRP), page 3-8	1. HSRP Basic Functionality (ACE Setup)	
Layer 3 Protocols, page 3-8	Open Shortest Path First (OSPF), page 3-9	1. OSPF Database Verification (ACE Setup) 2. OSPF Route Summarization (ACE Setup)	

Table 3-1 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Negative, page 3-12	Hardware Failure, page 3-12	1. Failure of Etherchannel Module on dca-agg-1 (ACE Setup) 2. Failure of Etherchannel Module on dca-agg-2 (ACE Setup) 3. Reset of Aggregation Layer Device dca-agg-1 (ACE Setup) 4. Reset of Aggregation Layer Device dca-agg-2 (ACE Setup)	
Negative, page 3-12	Link Failure, page 3-18	1. Failure of 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2 (ACE Setup) 2. Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1 (ACE Setup) 3. Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1 (ACE Setup) 4. Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2 (ACE Setup) 5. Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2 (ACE Setup) 6. Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1 (ACE Setup) 7. Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2 (ACE Setup) 8. Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1 (ACE Setup) 9. Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2 (ACE Setup) 10. Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2 (ACE Setup) 11. Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1 (ACE Setup) 12. Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2 (ACE Setup) 13. Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2 (ACE Setup)	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Baseline

The baseline tests are focused on various aspects of administering the devices in the DCAP test topology, as well as the verification of the most basic features such as distributed forwarding and security. Baseline tests verify network is in working order prior to starting testing and quantify steady state network performance.

This section contains the following topics:

Baseline

In all of DCAP testing, system resources of all the Layer 2/3 devices in the test topology are monitored, including CPU and memory utilization. When an issue is suspected, manifest as a sustained CPU spike or consumed memory for example, it is helpful to have a steady-state baseline of what the network resources look like for comparison purposes. The tests in this section help to establish a baseline level of expected behavior so that real problems can be more easily identified.

This section contains the following topics:

Topology Baseline (ACE Setup)

This test verified that the network is in an operational state. An initial snapshot of the current network state is taken. Background traffic is left running for approximately two hours. At the end of this time the current network state is compared to the baseline snapshot taken at the beginning of this test. This comparison verifies the network is ready for testing by examining features, protocols and interface counters for discrepancies.

It also provides a baseline of what the system resources (CPU and memory) look like while the traffic that is used in the tests is running. This is useful for comparison purposes during the other tests.

Test Procedure

The procedure used to perform the Topology Baseline (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Baseline all CDP neighbor relationships. Run the CDP crawler script verifying all expected CDP neighbors are reported.

The purposed of the CDP crawler script is to crawl through the network continuously, noting any changes that occur between traversals in CDP information. It parses information gathered from select CDP and IOS commands.

Step 3 Baseline all EtherChannel members. Run the channel crawler script verifying that all interfaces expected to be in channels are reported.

The purpose of the channel crawler script is to run through a network and verify that EtherChannels are in a proper state. It parses information gathered from select EtherChannel and IOS commands.

Step 4 Baseline all trunk interfaces. Run the trunk crawler script verifying that all expected trunking interfaces, configuration, and status are reported.

The purpose of the trunk crawler script is to run through a network and verify that trunking is in a proper state. It parses information gathered from select trunking and IOS commands.

Step 5 Baseline all interface states and counters. Run the interface crawler script recording interface counters and states.

The interface crawler script crawls through a network continually. All up/up interfaces are checked for various errors. Initially all non zero error counters will be logged, then any counters that increment from that point on.

Step 6 Baseline all interface UDLD states. Run the UDLD crawler script recording the UDLD state of all interfaces.

The UDLD crawler script gathers a list of UDLD ports from a list of devices and traverses their neighbors continuously, checking for UDLD problems or inconsistencies. It parses information gathered from select UDLD and IOS commands.

Step 7 Baseline all linecards used in the topology. Run the module crawler script recording module counters and state.

The module crawler script gathers a list of modules from a list of devices and looks for problems or inconsistencies. It parses information gathered from select module and IOS commands.

Step 8 Begin the test traffic. Allow it to run for two hours.

Step 9 Execute the CDP crawler script to verify that the CDP feature is operating in the Data Center test network as it was before background traffic was started.

Step 10 Execute the channel crawler script to verify that the EtherChannel feature is operating in the Data Center test network as it was before background traffic was started.

Step 11 Execute the trunk crawler script to verify that the trunking feature is operating in the Data Center test network as it was before background traffic was started.

Step 12 Execute the interface crawler script to verify that the basic functionality of the interface is operating in the Data Center test network as it was before background traffic was started.

Step 13 Execute the UDLD crawler script to verify that the UDLD feature is operating in the Data Center test network as it was before background traffic was started.

Step 14 Execute the module crawler script to verify that the line cards in the Data Center test network are still operating correctly after background traffic was started.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that there will be no change in the test topology during the baseline period.
- We expect no CPU or memory problems.

Results

Topology Baseline (ACE Setup) passed.

Device Management

Simple Network Management Protocol (SNMP) is an application-layer protocol designed to facilitate the exchange of management information between network devices. The tests in this section cover some of the common procedures and features used in the normal operation of a network, including the upgrading of network devices and the use of various features that may be used in troubleshooting.

This section contains the following topics:

- General On-Line Diagnostics (GOLD) (ACE Setup), page 3-5

General On-Line Diagnostics (GOLD) (ACE Setup)

General online diagnostics (GOLD) is a software tool that tests and verifies the hardware functionality of the supervisor engine, modules, and switch while the switch is connected to a live network. There are disruptive and non disruptive online diagnostic tests, including a subset of the GOLD tests that are run upon bootup of a hardware component. These are referred to as bootup diagnostics and are run during bootup, module OIR, or switchup to a redundant supervisor.

Each device in the data center topology is configured for a complete diagnostics run on bootup. This test verifies that each device in the data center topology is configured to run complete diagnostics on bootup, and that the complete set of diagnostics was run on each module at the last boot event.

Test Procedure

The procedure used to perform the General On-Line Diagnostics (GOLD) (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Log into dca-core-1 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 3 On dca-core-1, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 4 Log into dca-core-2 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 5 On dca-core-2, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 6 Log into dca-agg-1 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 7 On dca-agg-1, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 8 Log into dca-agg-2 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 9 On dca-agg-2, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 10 Log into dca-acc-6k-1 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 11 On dca-acc-6k-1, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 12 Log into dca-acc-6k-2 and use the **show diagnostic bootup level** command to verify that the current level is set to complete.

The current diagnostic bootup level should be complete.

Step 13 On dca-acc-6k-2, use the **show diagnostic result all** command to verify that the complete set of diagnostics was run against each module in the box, and that there were no failed tests.

There should be no tests with a result marked F, or failed.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the complete set of online diagnostics to have run on all modules in the systems under test, as configured.

- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

General On-Line Diagnostics (GOLD) (ACE Setup) passed.

Traffic Forwarding

This section of tests measures some of the basic traffic forwarding features and abilities of the DCAP test topology.

This section contains the following topics:

Zero Packet Loss (ACE Setup)

This test verified that the network devices in the Data Center topology are able to forward basic network traffic, without loss, in a steady-state condition. Web (HTTP/HTTPS) traffic consisting of varying frame sizes is sent between client devices and web servers. No negative, or failure, events are introduced during this test. The network devices will all be monitored for errors, and for CPU and memory usage stability.

Test Procedure

The procedure used to perform the Zero Packet Loss (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin the background test traffic that will send 30 minutes' worth of HTTP, HTTPS, and FTP traffic between the clients and the servers.

Step 3 When the traffic completes, measure the percentage of connection attempts that resulted in error codes. This percentage should be less than one percent.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss experienced by the background test traffic to be within tolerances.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Zero Packet Loss (ACE Setup) passed.

Layer 3 Protocols

This section of the test plan looks at the functionality of some of the common Layer 3 protocols used in the DCAP test topology.

This section contains the following topics:

Hot Standby Router Protocol (HSRP)

Hot Standby Router Protocol (HSRP) is used to provide a redundant gateway IP address to clients on a particular subnet. In the DCAP test topology, the virtual IP address (gateway) is shared by two routers, dca-agg-1 and dca-agg-2. Each of these two routers is configured with two IP addresses per HSRP subnet, one that is unique to that router, and one that is shared with the peer HSRP router. The router with the highest HSRP priority will assume Active state and respond to queries on the Virtual IP. The other router will assume Standby state and ignore such queries, while in Standby state.

This section contains the following topics:

HSRP Basic Functionality (ACE Setup)

Hot Standby Router Protocol (HSRP) is used to provide a redundant gateway IP address to clients on a particular subnet. In the DCAP test topology, the virtual IP address (gateway) is shared by two routers, dca-agg-1 and dca-agg-2. Each of these two routers is configured with two IP addresses per HSRP subnet, one that is unique to that router, and one that is shared with the peer HSRP router. The router with the higher HSRP priority will assume Active state and respond to queries on the Virtual IP. The other router will assume Standby state and ignore such queries while in Standby state.

There are 200 HSRP groups in the DCAP test topology, providing virtual gateways for over 200 subnets. This test verified that the Aggregation Layer devices were able to scale to this number of standby groups. It verified that the correct router was in Active HSRP state and that only that router was displaying the HSRP MAC address.

Test Procedure

The procedure used to perform the HSRP Basic Functionality (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the HSRP configuration for VLAN's 1101 to 1300 on dca-agg-1 and dca-agg-2 using the **show running-config | begin interface Vlan1101** command.

On dca-agg-1, these 200 VLAN's are configured with a standby priority of 120. On dca-agg-2, they are configured with a standby priority of 110. Each VLAN has a standby IP address, and each belongs to a separate standby group (specified by the number directly following standby).

Step 3 Use the **show standby brief** command to verify that dca-agg-1 is active for VLAN's 1101 to 1300 and that dca-agg-2 is standby.

Step 4 Verify that dca-agg-1 has a virtual MAC address running on each of the standby VLAN's for which it is the active HSRP router using the **show standby | include Vlan|Virtual mac**.

Each VLAN has a virtual MAC address assigned to it.

Step 5 Verify that dca-agg-2 does not have a virtual MAC address running on the standby VLAN's for which it is the standby HSRP router using the **show standby | include Vlan|Virtual mac**.

None of the VLAN's have a virtual MAC address assigned to it.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all HSRP groups to reflect their configuration in their active and standby states.
- We expect all active HSRP groups to have an associated virtual MAC address and that no standby HSRP groups will have a MAC address.
- We expect no CPU or memory problems.

Results

HSRP Basic Functionality (ACE Setup) passed.

Open Shortest Path First (OSPF)

Open Shortest Path First (OSPF) is an Interior Gateway Protocol (IGP) developed by the OSPF working group of the Internet Engineering Task Force (IETF). Designed expressly for IP networks, OSPF supports IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets.

This section contains the following topics:

OSPF Database Verification (ACE Setup)

Each Layer 3 device in an autonomous system running OSPF maintains a detailed view, or database, of the entire OSPF network. This database contains information gathered about networks that have been advertised via OSPF, and OSPF routers in the network. Information about networks and OSPF routers is propagated through the OSPF network using several different types of Link State Algorithm (LSA) messages.

This test verified that the OSPF database contains the information that would be expected for this particular test topology.

Test Procedure

The procedure used to perform the OSPF Database Verification (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 From each router in the DCAP test topology that is running OSPF, gather information about the OSPF interfaces and the OSPF processes that are running using the **show ip ospf database**, **show ip ospf database database-summary**, and **show ip ospf interface** commands.

The devices in the DCAP test topology that are running OSPF include dca-agg-1, dca-agg-2, dca-core-1, dca-core-2, and dca-client.

Step 3 Verify that the number of Router (Type 1) LSA's in each area is what is expected.

For each router in a particular area, there should be a single router LSA given.

Step 4 Verify that the number of Network (Type 2) LSAs in each area is what is expected.

A Type 2 LSA is generated when a network segment exists that has both a DR and a BDR. In other words, when a network shared by two devices are both running OSPF on that network, a network LSA is generated. The number of Network LSA's is determined by how many of those types of links are in a given area.

Step 5 Verify that the number of Summary Network (Type 3) LSA's in each area is what is expected.

The ABR, or the router that is at the border of two or more areas, sends a summary network LSA into Area X for all other ares other than X for which it is a participant. So if you have Areas X, Y, and Z and RouterA is a participant in all of them, it will generate a Type-3 LSA and send it into Area X. This LSA will contain a summary of all the networks found in Areas Y and Z.

Step 6 Verify that the number of Summary ASBR (Type 4) LSAs in each area is what is expected.

An ABR between areas X and Y will generate a Summary ASBR into X for an ASBR that exists in Y. However, an ABR which is an ASBR will not advertise itself. In each area, verify there is a Summary ASBR LSA from each ABR in that area for each ASBR outside this area.

Step 7 Verify that the number of AS-External (Type 5) LSA's in each area is what is expected.

This is pretty much the number of networks that are being redistributed from another routing protocol into OSPF times the number of ABR's that share redistribution responsibilities.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the OSPF database to be populated with the correct number of LSA's of each type for the topology being used.

Results

OSPF Database Verification (ACE Setup) passed.

OSPF Route Summarization (ACE Setup)

In the DCAP test topology, in OSPF Area 10, there are several /30 subnets configured for the inter-switch links. These all share a 172.31.1.x prefix. All servers in Area 10 are on /24 subnets and share a prefix of 101.1.x.x.

The Core routers in the Data Center test topology, dca-core-1 and dca-core-2, as part of the default configuration, summarize the subnets in Area 10 for advertisement into Area 0. The 172.31.1.x/30 subnets are configured to summarize as 172.31.1.0/24 networks while the 101.1.x.x/24 subnets are configured to summarize as 101.1.0.0/16 networks.

This test verified that summarization was occurring by looking at the routing table of a device in Area 20. The memory and CPU utilization were monitored for stability.

Test Procedure

The procedure used to perform the OSPF Route Summarization (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Use the **show running-config | begin router ospf** command on dca-core-1 and dca-core-2 to verify that summarization is configured.

Each of the devices will have the following two lines as part of their OSPF configuration:

area 10 range 101.3.0.0 255.255.0.0area 10 range 172.31.1.0 255.255.255.0

The first line will cause any networks in Area 10 matching 101.1.x.x to be summarized into a /16. The second line will cause any networks in Area 10 matching 172.31.1.x to be summarized into a /24.

Step 3 On dca-agg-1, use the **show running-config interface Te9/2** to verify that this device has an interface with a /30 address matching the 172.31.1.x format.

The IP address of interface Te9/2 is 172.31.1.14/30.

Step 4 On dca-client, use the **show ip route 172.31.1.14** command to verify that the route to this address has been summarized as a /24.

The output of this command will show a "Routing entry for 172.31.1.0/24".

Step 5 On dca-agg-1, use the **show running-config interface Vlan1101** to verify that this device has an interface with a /24 address matching the 101.3.x.x format.

The IP address of interface Vlan1101 is 101.3.1.2/24.

Step 6 On dca-client, use the **show ip route 101.3.1.2** command to verify that the route to this address has been summarized as a /16.

The output of this command will show a "Routing entry for 101.3.0.0/16".

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the appropriate networks will be summarized correctly by the OSPF Area Border Routers.

- We expect no CPU or memory problems.

Results

OSPF Route Summarization (ACE Setup) passed.

Negative

The suite of negative tests are aimed at measuring the response of the test topology to various negative events, such as simulated hardware failures, link failures and whole device failures.

This section contains the following topics:

Hardware Failure

This group of tests measures the ability of the test topology to absorb various hardware failures, including system crashes and module resets.

This section contains the following topics:

Failure of Etherchannel Module on dca-agg-1 (ACE Setup)

The port-channel between the two Aggregation Layer device, dca-agg-1 and dca-agg-2, is critical for the monitoring of health and the synchronization of connections between the service modules. This link, an 802.1q trunk, carries the VLAN's that communicate the heartbeat messages between the CSM's and the FWSM's. Further, it replicates the connection states between the peer service modules so that downtime due to failover is minimized. The redundancy of this port-channel is essential.

In the DCAP test topology, this port-channel comprises two TenGigabitEthernet links, each link on a separate WS-X6704-10GE module. This test verified that if one of these modules were to reset, the impact to traffic would be minimal. Each of the two modules will be flapped multiple times. Client-to-server TCP traffic will be monitored to verify that there are no adverse effects.

Test Procedure

The procedure used to perform the Failure of Etherchannel Module on dca-agg-1 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 4 Use the **show module** command to verify that the WS-X6704-10GE modules in slots 9 and 10 are online.

Step 5 On dca-agg-1, reset the TenGigabitEthernet module in slot 9 using the **hw-module module 9 reset** command.

Step 6 When module 9 comes back online, verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the **show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 7 Repeat the reset of module 9 on dca-agg-1 nine times for a total of ten flaps.

Step 8 Measure any traffic loss due to the module being reset.

Step 9 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 still consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 10 Use the **show module** command to verify that the WS-X6704-10GE modules in slots 9 and 10 are online.

Step 11 On dca-agg-1, reset the TenGigabitEthernet module in slot 10 using the **hw-module module 10 reset** command.

Step 12 When module 10 comes back online, verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 13 Repeat the reset of module 10 on dca-agg-1 nine times for a total of ten flaps.

Step 14 Measure any traffic loss due to the module being reset.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the port-channel interface to maintain normal operation as a logical interface when one of the hardware modules is reloaded.
- We expect any traffic loss due to the module reload to be minimal.

- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of Etherchannel Module on dca-agg-1 (ACE Setup) passed.

Failure of Etherchannel Module on dca-agg-2 (ACE Setup)

The port-channel between the two Aggregation Layer device, dca-agg-1 and dca-agg-2, is critical for the monitoring of health and the synchronization of connections between the service modules. This link, an 802.1q trunk, carries the VLAN's that communicate the heartbeat messages between the CSM's and the FWSM's. Further, it replicates the connection states between the peer service modules so that downtime due to failover is minimized. The redundancy of this port-channel is essential.

In the DCAP test topology, this port-channel is composed of two TenGigabitEthernet links, each link on a separate WS-X6704-10GE module. This test verified that if one of these modules were to reset, the impact to traffic would be minimal. Each of the two modules will be flapped multiple times. Client-to-server TCP traffic will be monitored to verify that there are no adverse effects.

Test Procedure

The procedure used to perform the Failure of Etherchannel Module on dca-agg-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 4 Use the **show module** command to verify that the WS-X6704-10GE modules in slots 9 and 10 are online.

Step 5 On dca-agg-2, reset the TenGigabitEthernet module in slot 9 using the **hw-module module 9 reset** command.

Step 6 When module 9 comes back online, verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 7 Repeat the reset of module 9 on dca-agg-2 nine times for a total of ten flaps.

Step 8 Measure any traffic loss due to the module being reset.

Step 9 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 still consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 10 Use the **show module** command to verify that the WS-X6704-10GE modules in slots 9 and 10 are online.

Step 11 On dca-agg-2, reset the TenGigabitEthernet module in slot 10 using the **hw-module module 10 reset** command.

Step 12 When module 10 comes back online, verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 13 Repeat the reset of module 10 on dca-agg-2 nine times for a total of ten flaps.

Step 14 Measure any traffic loss due to the module being reset.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the port-channel interface to maintain normal operation as a logical interface when one of the hardware modules is reloaded.
- We expect any traffic loss due to the module reload to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of Etherchannel Module on dca-agg-2 (ACE Setup) passed.

Reset of Aggregation Layer Device dca-agg-1 (ACE Setup)

The Aggregation Layer provides the bulk of services for traffic coming into the data center. Services such as server load balancing, SSL decryption and encryption, and firewalling are provided by the Aggregation Layer devices dca-agg-1 and dca-agg-2.

Redundancy in the aggregation layer is important for providing a high level of availability for these services. The DCAP test topology was designed to provide redundancy through a pair of Aggregation Layer devices rather than redundant supervisors because the failover timers for many of the services are set low. This means that a service failover could be triggered even by a fast SSO/NSF failover. This is inline with the goals of predictability with regards to traffic in the data center. If anything less than all services fail over, the result is an unclean picture of traffic using both Aggregation Layer boxes for the various services before arriving at the destination.

By configuration, dca-agg-1 is the primary provider of services in the DCAP test topology. It is the STP root, the Active HSRP router, and it houses the active CSM and FWSM. The standby services wait on dca-agg-2 for a failover event.

This test verified the impact on traffic due to the failure of the Aggregation device dca-agg-1. Background traffic was run using Linux servers and measurable traffic using the Shenick test tool. A total of five system resets were performed.

Test Procedure

The procedure used to perform the Reset of Aggregation Layer Device dca-agg-1 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Start the background traffic and the measurable Shenick traffic.

Step 3 Once traffic is running, verify that dca-agg-1 is passing traffic through interfaces Te9/1, Te9/2, Te9/4, Te10/1, Te10/2, and Te10/4 using the **show interfaces counters | include Port|Te9|Te10** command.

Step 4 Use the **show interfaces counters | include Port|Te9|Te10** command to verify that dca-agg-2 is not passing any substantial traffic.

There will be a variable amount of management traffic being passed by dca-agg-2.

Step 5 Reload dca-agg-1.

Step 6 Use the **show interfaces counters | include Port|Te9|Te10** command to verify that dca-agg-2 is now passing traffic.

Step 7 When dca-agg-1 comes back online, verify that it is passing traffic through interfaces Te9/1, Te9/2, and Po1 using the **show interfaces counters | include Port|Te9|Te10|Po1** command.

The command in this step asks to look at the traffic counters on Port-channel 1 as well. This is because, following the recovery of dca-agg-1, the FWSM will remain active in dca-agg-2. The version of FWSM code that is running in DCAP Phase One does not have a preempt feature, and so the FWSM in dca-agg-1 never resumes the active role. This is why below, a manual reset of the FWSM in dca-agg-2 is called for.

This is also the reason that no traffic is seen from dca-agg-1 down to the Access Layer switches through interfaces Te9/4, Te10/1, Te10/2, or Te10/4. In order for traffic to pass between client and server, it must go through the active FWSM, which is now in dca-agg-2 (and will remain so indefinitely).

Step 8 Use the **show interfaces counters | include Port|Te9|Te10|Po1** command to verify that dca-agg-2 is passing traffic over Port-channel 1 and the interfaces that connect to the downstream Access Layer devices, Te9/4, Te10/1, Te10/2, and Te10/4.

Step 9 Reboot the FWSM in dca-agg-2 using the **hw-module module 1 reset** command.

This will force the FWSM in dca-agg-1 back into active mode, the starting point for this failover test.

Step 10 Verify that dca-agg-1 is again passing traffic through interfaces Te9/4, Te10/1, Te10/2, and Te10/4 using the **show interfaces counters | include Port|Te9|Te10|Po1** command.

Step 11 Repeat the above sequence of reloading the DUT and checking counters four times.

Step 12 Verify the traffic that was lost via the Shenick test tool and the Linux-generated tool.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect there to be traffic loss on the order of five seconds or less for each failover performed.
- We expect the reset device comes back online ande resume forwarding traffic as before.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Reset of Aggregation Layer Device dca-agg-1 (ACE Setup) passed.

Reset of Aggregation Layer Device dca-agg-2 (ACE Setup)

The Aggregation Layer provides the bulk of services for traffic coming into the data center. Services such as server load balancing, SSL decryption and encryption, and firewalling are provided by the Aggregation Layer devices dca-agg-1 and dca-agg-2.

Redundancy in the aggregation layer is important for providing a high level of availability for these services. The DCAP test topology was designed to provide redundancy through a pair of Aggregation Layer devices rather than redundant supervisors because the failover timers for many of the services are set very low. This means that a service failover could be triggered even by a fast SSO/NSF failover. This is in-line with the goals of predictability with regards to traffic in the data center. If anything less than all services fail over, the result is a unclean picture of traffic using both Aggregation Layer boxes for the various services before arriving at the destination.

By configuration, dca-agg-2 is the standby provider of services in the DCAP test topology. It is the STP secondary root, the standby HSRP router, and it houses the standby CSM and FWSM. These standby services wait on dca-agg-2 for a failover event.

This test verified the impact on traffic due to the failure of the Aggregation device dca-agg-2. Background traffic was run using Linux servers and measurable traffic using the Shenick test tool. A total of five system resets were performed.

Test Procedure

The procedure used to perform the Reset of Aggregation Layer Device dca-agg-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Start the background traffic and the measurable Shenick traffic.

Step 3 Once traffic is running, verify that dca-agg-2 is not passing any significant data traffic through interfaces Te8/1, Te8/2, Te9/4, Te9/1, Te9/2, Po 1 using the **show interface | include packets/sec** command.

The four TenGigabit Ethernet interfaces are connected downstream to the Access Layer switches. Because dca-agg-2 is not providing any active services for data center traffic, other than SSL decryption/encryption, no data traffic should be transiting dca-agg-2. The EtherChannel Po1 connects dca-agg-2 to dca-agg-1 and, in steady-state, is used solely for management traffic.

Note that the output will show significant traffic being sent to the Access Layer devices. This is traffic from the Shenick test tool that is being flooded as a result of the tool not answering periodic ARP requests.

Step 4 Reload dca-agg-2.

Step 5 When dca-agg-2 comes back online, verify that it is, again, not passing any substantial traffic using the **show interfaces counters | include Port|Te9|Te10|Po1** command.

Note that in the output, it will show significant traffic being sent to the Access Layer devices. This is traffic from the Shenick test tool that is being flooded as a result of the tool not answering periodic ARP requests.

Step 6 Repeat the above sequence of reloading dca-agg-2 and checking counters four times.

Step 7 Verify the traffic that was lost via the Shenick test tool and the Linux-generated tool.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect there to traffic loss on the order of three seconds or less for each failover performed.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Reset of Aggregation Layer Device dca-agg-2 (ACE Setup) passed.

Link Failure

The tests included in this section measure the impact of a link failure occurring in the data path. The ability of the data center infrastructure to respond favorable to such scenarios is critical to the high availability of network resources.

This section contains the following topics:

- Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2 (ACE Setup), page 3-28

Failure of 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Access Layer device, dca-acc-4k-1, and an Access Layer device, dca-acc-4k-2. Web traffic was sent using a test tool to a VIP on the CSM. Te1/50 on dca-acc-4k-1 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-acc-4k-1 and shut down interface Te1/50.

Step 4 After two minutes, bring interface Te1/50 back online using the no shutdown command on dca-acc-4k-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2 (ACE Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and an Access Layer device, dca-acc-4k-1. Web traffic was sent using a test tool to a VIP on the CSM. Te9/6 on dca-agg-1 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-1 and shut down interface Te9/6.

Step 4 After two minutes, bring interface Te9/6 back online using the no shutdown command on dca-agg-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1 (ACE Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and an Access Layer device, dca-acc-6k-1. Web traffic was sent using test tool to a VIP on the CSM. Te9/4 on dca-agg-1 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-1 and shut down interface Te9/4.

Step 4 After two minutes, bring interface Te9/4 back online using the no shutdown command on dca-agg-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1 (ACE Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and an Access Layer device, dca-acc-6k-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te10/1 on dca-agg-1 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-1 and shut down interface Te10/1.

Step 4 After two minutes, bring interface Te10/1 back online using the **no shutdown** command on dca-agg-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2 (ACE Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and an Access Layer device, dca-acc-4k-2. Web traffic was sent using a test tool to a VIP on the CSM. Te8/2 on dca-agg-2 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-2 and shut down interface Te8/2.

Step 4 After two minutes, bring interface Te8/2 back online using the no shutdown command on dca-agg-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2 (ACE Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and an Access Layer device, dca-acc-6k-1. Web traffic was sent using a test tool to a VIP on the CSM. Te9/4 on dca-agg-2 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-2 and shut down interface Te9/4.

Step 4 After two minutes, bring interface Te9/4 back online using the no shutdown command on dca-agg-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1 (ACE Setup) passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and an Access Layer device, dca-acc-6k-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te10/1 on dca-agg-2 was shutdown for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-agg-2 and shut down interface Te10/1.

Step 4 After a minute, bring interface Te10/1 back online using the **no shutdown** command on dca-agg-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2 (ACE Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1 (ACE Setup)

This test measured the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and a Core Layer device, dca-core-1. Web traffic is sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/2 on dca-core-1 is shut down for two minutes then brought back online for two minutes. This is repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-1 and shut down interface Te1/2.

Step 4 After a minute, bring interface Te1/2 back online using the **no shutdown** command on dca-core-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1 (ACE Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and a Core Layer device, dca-core-1. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/3 on dca-core-1 was shut down for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-1 and shut down interface Te1/3.

Step 4 After a minute, bring interface Te1/3 back online using the **no shutdown** command on dca-core-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.

- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2 (ACE Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between the two Core Layer devices, dca-core-1 and dca-core-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/1 on dca-core-1 was shut down for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-1 and shut down interface Te1/1.

Step 4 After a minute, bring interface Te1/1 back online using the **no shutdown** command on dca-core-1.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2 (ACE Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-1, and a Core Layer device, dca-core-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/2 on dca-core-2 was shut down for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-2 and shut down interface Te1/2.

Step 4 After a minute, bring interface Te1/2 back online using the **no shutdown** command on dca-core-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1 (ACE Setup) passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2 (ACE Setup)

This test verified the impact on traffic from a failure of the TenGigabitEthernet link between an Aggregation Layer device, dca-agg-2, and a Core Layer device, dca-core-2. Web traffic was sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/3 on dca-core-2 was shut down for two minutes then brought back online for two minutes. This was repeated for a total of 10 interface flaps.

Test Procedure

The procedure used to perform the Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Log in to dca-core-2 and shut down interface Te1/3.

Step 4 After a minute, bring interface Te1/3 back online using the **no shutdown** command on dca-core-2.

Step 5 Repeat the interface flap nine times for a total of ten flaps.

Step 6 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2 (ACE Setup) passed.

Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2 (ACE Setup)

The port-channel between the two Aggregation Layer device, dca-agg-1 and dca-agg-2, is critical for the monitoring of health and the synchronization of connections between the service modules. This link, an 802.1q trunk, carries the VLAN's that communicate the heartbeat messages between the CSM's and the FWSM's. Further, it replicates the connection states between the peer service modules so that downtime due to failover is minimized. The redundancy of this port-channel is essential.

This test verified that if a single link of that port-channel were to go down, the impact to traffic would be minimal. Each of the two TenGigabitEthernet links in the port-channel were flapped multiple times. Client-to-server TCP traffic was monitored to verify that there were no adverse effects.

Test Procedure

The procedure used to perform the Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2 (ACE Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin sending HTTP test traffic using the Shenick test tool.

Step 3 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 4 On dca-agg-1, shut down interface Te9/3.

Step 5 After a minute, bring interface Te9/3 back online using the **no shutdown** command on dca-agg-1.

Step 6 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 7 Repeat the flap of interface Te9/3 on dca-agg-1 nine times for a total of ten flaps.

Step 8 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 9 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 still consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 10 On dca-agg-1, shut down interface Te10/3.

Step 11 After a minute, bring interface Te10/3 back online using the **no shutdown** command on dca-agg-1.

Step 12 Verify that the Port-channel 1 interface on dca-agg-1 and dca-agg-2 again consists of two bundled TenGigabitEthernet interfaces, and that they are active using the**show etherchannel 1 summary** command.

Two interfaces, Te9/3 and Te10/3, should be listed as ports belonging to Po1. They should each have a P-flag next to them (in parentheses), indicating that they are bundled in the port-channel. This applies to the output seen from each device.

Step 13 Repeat the flap of interface Te10/3 on dca-agg-1 nine times for a total of ten flaps.

Step 14 Measure any traffic loss due to the interface being shut down and being brought back online.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the port-channel interface to maintain normal operation as a logical interface when a single bundled link is flapped.
- We expect any traffic loss due to the link failure to be minimal.
- We expect any traffic loss due to the link recovery to be minimal.

- We expect no CPU or memory problems.

Results

Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2 (ACE Setup) passed.

CHAPTER 4

Layer 4-7 CSM

Tests in the "Layer 4-7 CSM" section on page 4-1 focus on traffic flows that touch multiple service modules. The service module combination that was used for this section was CSM+FWSM+SSLM+IDSM+NAM. Figure 1-1 illustrates the relevant topology for this section.

Refer to the following chapters for respective Layer 4-7 services testing:

- ACE—Based Integrated Switch Bundle "Layer 4-7 ACE" section on page 3-1
- CSM—Based Service Chassis Bundle "Layer 4-7 Services Switch" section on page 4-1
- Application Control Engine (ACE) "ACE" section on page 5-1
- Intrusion Detection Services Module (IDSM) "IDSM IPS" section on page 6-1

Test Results Summary

Table 4-1 on page 4-2 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 4-1 on page 4-2 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 4-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
CSM FWSM, page 4-3	n/a	1. Active FTP Through FWSM and CSM (CSM Setup) 2. DNS Query Through CSM and FWSM 3. FWSM and CSM Layer 4 SYN Attack (CSM Setup) 4. ICMP to a CSM Layer 3 and Layer 4 Vserver (CSM Setup) 5. Idle Timeout UDP (CSM Setup) 6. Passive FTP Through FWSM and CSM (CSM Setup)	CSCsl39483
CSM SSLM Focused, page 4-16	n/a	1. URL Rewrite (CSM Setup)	
Redundancy, page 4-20	n/a	1. CSM Redundancy Test (CSM Setup) 2. FWSM Redundancy (CSM Setup) 3. HSRP Failover (CSM Setup) 4. SSLM Reset (CSM Setup)	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

CSM FWSM

The tests in this section look at some of the interoperability capacities of the CSM and FWSM, in terms of how they work together to handle data traffic.

Active FTP Through FWSM and CSM (CSM Setup)

This test verified that the FWSM and CSM properly handled active FTP traffic when the **ftp fixup protocol 21** and was enabled and disabled on the FWSM. FTP traffic was sent from an outside client to vserver VIP-ACTIVE-FTP and from an inside client to an outside server. In version 3.x of the FWSM code the ftp fixup protocol command has been superceeded by the Inspection engine. If you disable FTP inspection engines with the no inspect ftp command, outbound users can start connections only in passive mode, and all inbound FTP is disabled.

Relevant CSM Configuration:

```
 real P1_ETH1.3001
  address 101.1.1.11
  location dca-penguin-1
  inservice
 real P1_ETH1.3002
  address 101.1.1.12
  location dca-penguin-1
  inservice
!
 serverfarm FARM1-A
  nat server
  no nat client
  predictor leastconns
  real name P1_ETH1.3001
   inservice
  real name P1_ETH1.3002
   inservice
!
 vserver VIP-ACTIVE-FTP
```

```
  virtual 101.40.1.251 tcp ftp
  vlan 301
  serverfarm FARM1-A
  advertise active
  replicate csrp connection
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the Active FTP Through FWSM and CSM (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following commands on the FWSM to clear connections and counters in the VLAN 1101-2101 context:

```
change context Vlan1101-2101
clear xlate
clear conn
clear access-list ACL-in count
clear logging buffer
```

Step 3 Issue the following commands on the FWSM to verify connections and counters have been cleared:

```
change context Vlan1101-2101
show xlate
show conn
show access-list ACL-in
```

Step 4 Issue the following commands on the active CSM to clear connections and counters:

```
clear mod csm 2 counters
clear mod csm 2 connections
```

Step 5 Issue the following commands on the active CSM to verify the counters have been cleared:

```
show mod csm 2 vserver name vip-active-ftp detail
show mod csm 2 real sfarm farm1-a detail
show mod csm 2 stats
show mod csm 2 conns
```

Step 6 Send active FTP traffic to vserver VIP-ACTIVE-FTP from an outside client.

Step 7 Issue the following commands on the FWSM to verify the FTP control and data channels were successfully created:

```
change context Vlan1101-2101
show xlate
show conn
show log
```

Step 8 Issue the **show mod csm 2 conns** command to verify the FTP control and data connections have been established.

Step 9 When the FTP traffic has completed issue the following command on the FWSM to verify a match on the correct access list:

```
show access-list ACL-in | include extended permit tcp any 101.1.1.0 255.255.255.0 eq ftp
```

Step 10 Issue the following command on the active CSM to verify the FTP traffic was properly load balanced:

```
show mod csm 2 vserver name vip-active-ftp detail
show mod csm 2 real sfarm farm1-a detail
show mod csm 2 stats
```

Step 11 On the FWSM context VLAN 1101-2101, configure the **no fixup protocol ftp 21** command.

The **fixup protocol ftp 21** configuration is part of the default configuration for the DCAP test topology.

Step 12 Send an active FTP request from an inside client to an outside server.

This connection should fail. When the **no fixup protocol ftp 21** command has been configured only passive mode FTP is allowed from an inside interface to an outside interface.

Step 13 Issue the following commands on the FWSM to verify the FTP data channel was not successfully created:

```
change context Vlan1101-2101
show xlate
show conn
show log
```

Step 14 Reconfigure the **fixup protocol ftp 21** command on the VLAN 1101-2101 context to enable the fixup protocol for FTP on port 21 and use the **show fixup protocol ftp** command to verify it is now been enabled.

Step 15 Issue the following commands on the FWSM to clear connections and counters:

```
change context Vlan1101-2101
clear xlate
clear conn
clear access-list ACL-in count
clear log
```

Step 16 Send active FTP traffic to vserver VIP-ACTIVE-FTP from an outside client.

Step 17 Issue the following commands on the FWSM to verify the FTP control and data channels were successfully created:

```
change context Vlan1101-2101
show xlate
show conn
show log
```

Step 18 Issue the **show mod csm 2 conns** command to verify the FTP control and data connections have been established.

Step 19 Stop background scripts to collect final status of network devices and analyze for error.

Step 20 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to permit active FTP on the outside interface.
- We expect the FWSM would deny active FTP on the inside to outside interface when fixup protocol FTP 21 is disabled in version 2.x
- If the FWSM code is 3.x then if you disable FTP inspection engines with the no inspect ftp command, outbound users can start connections only in passive mode, and all inbound FTP is disabled.
- We expect the CSM vserver to properly load balance active FTP.

Results

Active FTP Through FWSM and CSM (CSM Setup) passed.

DNS Query Through CSM and FWSM

This test verified that the FWSM and CSM properly handled DNS traffic when fixup protocol DNS was enabled. In this topology the CSM virtual is on the outside of the FWSM and the reals are on the inside of the FWSM.

Relevant CSM Configuration:

```
vserver VIP-DNS
  virtual 201.40.40.240 udp dns service per-packet
  vlan 301
  serverfarm FARM1
  advertise active
  persistent rebalance
  inservice
```

Relevant FWSM Access List:

access-list ACL-in extended permit udp any 201.1.1.0 255.255.255.0

Test Procedure

The procedure used to perform the DNS Query Through CSM and FWSM test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following commands on the active FWSM in context VLAN 1101-2101 to clear connections and counters:

- clear xlate
- clear conn
- clear access-list ACL-in count
- clear log

Step 3 Issue the **clear module csm 2 counters** and **clear module csm 2 connections** commands on the active CSM to clear connections and counters.

Step 4 Use Spirent Avalanche or Ixia to send a small set of DNS queries to vserver VIP-DNS

Step 5 Issue the **show xlate** command on the active FWSM to verify that a global entry was created for each real in serverfarm FARM1.

Step 6 Use the **show show access-list ACL-in** command to verify there are matches on the portion of the access list that permits UDP DNS queries.

The ACL line that permits this traffic is:

access-list ACL-in extended permit udp any 201.1.1.0 255.255.255.0

Step 7 Use the following commands on the active CSM to verify the DNS traffic was properly load balanced:

- show module csm 2 vserver name vip-dns detail
- show module csm 2 stats

The "total conns" should approximate the number of hits that was seen on the FWSM access-list.

Step 8 Use the **show module csm 2 real sfarm IXIAFARM-1 detail** command to verify that each real server in the serverfarm has made some connections.

The total conns established counter should be spread evenly accross the serverfarm

Step 9 Use the **clear module csm 2 counters** and **clear module csm 2 connections** commands on the active CSM to clear connections and counters.

Step 10 Use Spirent Avalanche or Ixia to send DNS queries to vserver VIP-DNS for domain name www.datacenter.com at rate of 1,000 users per second.

Step 11 While traffic is running issue the **show xlate** and **show conn | include most** commands on the VLAN 1101-2101 FWSM context to verify the xlate table and number of open connections.

You should see 65 global entries in the xlate table.

Step 12 Verify the number of attempts and number of failures on the Spirent Avalanche run tab or the Ixia stats screen

Step 13 Verify the DNS connections rate on the Spirent Avalanche client stats tab and the select DNS. DNS queries per second should be around 1,000. Or on the Ixia stats screen check for the number of users and connection rate

Step 14 Use the following commands on the active CSM to verify the DNS traffic was properly load balanced. Counters Tot matches and L4 Load-Balanced Decisions should have the same value. Verify the Tot matches counter equals the number of attempts on the Spirent Avalanche run tab.

```
show module csm 2 vserver name vip-dns detail
show module csm 2 stats
```

Step 15 Use the **clear module csm 2 counters** and **clear module csm 2 connections** commands on the active CSM to clear connections and counters.

Step 16 Use Spirent Avalanche or Ixia to send DNS queries to vserver VIP-DNS for domain name www.datacenter.com at rate of 1,500 users per second. At this rate we expect a large number of UDP request to fail/retries.

Step 17 While traffic is running use the following two commands to verify the xlate table and number of open connections. You should see 65 global entries in the xlate table.

```
show xlate
show conn | in most
```

Step 18 Verify the number of attempts and number of failures on the Spirent Avalanche run tab or on the Ixia stats screen

Step 19 Verify the DNS connections rate on the Spirent Avalanche client stats tab and the select DNS. DNS queries per second should be around 1,500 or Check the Ixia stats screen and verify the connection rate and number of users is 1,500

Step 20 Use the following commands on the active CSM to verify the DNS traffic was properly load balanced. Counters Tot matches and L4 Load-Balanced Decisions should have the same value. Verify the Tot matches counter equals the number of attempts on the Spirent Avalanche run tab or the Ixia stats screen

```
show mod csm 2 vserver name vip-dns detail
show mod csm 2 stats
```

Step 21 Stop background scripts to collect final status of network devices and analyze for error.

Step 22 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to permit DNS traffic to vserver DMZ2-DNS
- We expect the FWSM to NAT the DNS response to the outside client when fixup protocol DNS is enabled.
- We expect the FWSM not to NAT the DNS response to the inside client when fixup protocol DNS is enabled.
- We expect the CSM vserver to properly load balance DNS traffic.

Results

DNS Query Through CSM and FWSM passed.

FWSM and CSM Layer 4 SYN Attack (CSM Setup)

SYN-flood attacks aim at preventing a TCP/IP server from servicing request. The SYN flag is set in a TCP segment when a connection request is sent by a computer. The target server responds back with an ACK and waits for a response from the initiator. The SYN-flood attacker spoofs the source IP address so that the server never receives a response to the ACK. This causes the server to use up resources overloading the server and preventing it from responding to legitimate connection request.

TCP intercept protects inside systems from a DoS attack perpetrated by flooding an interface with TCP SYN packets. Enable this feature by setting the maximum embryonic connections option of the NAT and static commands.

When the embryonic limit has been surpassed, the TCP intercept feature intercepts TCP SYN packets from clients to servers on a higher security level. The TCP intercept feature implements software to protect TCP servers from TCP SYN-flooding attacks, which are a type of DoS attack. SYN cookies are used during the validation process and help to minimize the amount of valid traffic being dropped.

The embryonic limit is a feature that is enabled for any inbound connection (a connection that the FWSM considers from lower to higher security). For a connection to be inbound either hit a static or a global xlate.

This test verified the TCP intercept feature by sending one million SYN packets generated on a Linux server using random source IP addresses. The SYN packets were sent to a CSM Layer 4 server with 65 reals behind the FWSM.

Relevant CSM Configuration:

```
vserver VIP-WWW
  virtual 201.40.40.240 tcp www
  vlan 301
  serverfarm FARM1
  advertise active
  persistent rebalance
  inservice
```

Test Procedure

The procedure used to perform the FWSM and CSM Layer 4 SYN Attack (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Configure context VLAN 1101-2101 on the active FWSM with the following static command to enable the limitation of embryonic connections to 20.

```
static (inside,outside) 201.1.1.0 201.1.1.0  netmask 255.255.255.0 tcp 0 20You must also
clear xlate clear xlate
```

Step 3 Use the **clear module csm 2 counters** command to clear all CSM statistics and the **clear module csm 2 conn** command to clear all connections.

Step 4 Verify CSM utilization by using the **show module csm 2 tech-support utilization** command.

Step 5 On the FWSM system context, clear the Fast Path SYN Cookie Statistics Counters for NP-1 and NP-2 with the **clear np 1 syn** and **clear np 2 syn** commands.

Step 6 Verify CPU and memory utilization on the FWSM by using the **show cpu** and **show memory** commands from the system context.

Step 7 From the outside client send 10,000,000 SYN packets to vserver VIP-WWW with random source IP addresses.

Step 8 While the SYN attack traffic is being sent, verify the rate of the SYN attack on the FWSM by using the **show perfmon | inc TCP Intercept** command. Issue the command multiple times to obtain a good baseline.

Step 9 While SYN attack traffic is being sent, verify CSM utilization by using the **show module csm 2 tech-support utilization** command.

Step 10 Verify there are no errors on the CSM by using the following commands.

```
show mod csm 2 vserver name vip-www detail
show mod csm 2 reals sfarm farm1-a det
show mod csm 2 stats
```

Step 11 Verify the FWSM has issued a SYN cookie and verify the number of SYN packets intercepted by using the following commands.

```
show np 1 syn
show np 2 syn
```

Step 12 Verify FWSM CPU and memory utilization were not adversely impacted by the SYN attack by using the **show cpu** and **show memory** commands.

Step 13 Verify the FWSM log contains message number FWSM-6-106015 by using the **show log** command in context VLAN 1101-2101.

Step 14 Remove static statement from VLAN 1101-2101 on the active FWSM with the following command.

```
no static (inside,outside) 201.1.1.0 201.1.1.0  netmask 255.255.255.0 tcp 0 20
```

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to intercept SYN packets being sent to the CSM reals by using a SYN cookie.
- We expect CPU and memory utilization on the CSM and FWSM not to be adversely impacted by the SYN attack.
- We expect the CSM to evenly load balance packets across all reals in the serverfarm.

Results

FWSM and CSM Layer 4 SYN Attack (CSM Setup) failed. The following failures were noted: CSCsl39483.

ICMP to a CSM Layer 3 and Layer 4 Vserver (CSM Setup)

This test verified ICMP ping traffic to multiple Layer 4 vservers and a Layer 3 vserver all configured with the same virtual IP address. The CSM virtual address is located on the outside of the FWSM and the CSM reals are located on the inside of the CSM.

Relevant CSM Configuration:

```
!
 vserver DMZ1-FTP
  virtual 201.40.40.240 tcp ftp service ftp
  vlan 301
  serverfarm FARM1
  advertise active
  persistent rebalance
  inservice
!
vserver VIP-DNS
  virtual 201.40.40.240 udp dns service per-packet
  vlan 301
  serverfarm FARM1
  advertise active
  persistent rebalance
  inservice
!
 vserver VIP-L3
  virtual 201.40.40.240 any
```

```
  vlan 301
  serverfarm FARM1-A
  advertise active
  persistent rebalance
  inservice
!
 vserver VIP-WWW
  virtual 201.40.40.240 tcp www
  vlan 301
  serverfarm FARM1-A
  advertise active
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the ICMP to a CSM Layer 3 and Layer 4 Vserver (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the **clear module csm 2 counters** command to clear all CSM statistics.

Step 3 Issue the following commands on the active FWSM in context VLAN 1101-2101 to clear connections and counters.

```
clear xlate
clear conn
clear logging buffer
```

Step 4 Suspend CSM vserver VIP-L3 with the **no inservice** command.

Step 5 From an outside Linux client send ICMP ping to CSM vserver VIP-WWW. This ping should be successful.

Step 6 On the active FWSM issue the **show xlate** command.

Step 7 Verify the following vservers have not recorded any policy matches or packets received by using the following commands.

```
show module csm 2 vservers name DMZ1-FTP detail
show module csm 2 vservers name vip-dns detail
show module csm 2 vservers name vip-www detail
show module csm 2 vservers name vip-l3 detail
```

Step 8 Enable CSM vserver VIP-L3 with the **inservice** command and verify that it is now operational with the **show module csm 2 vserver vip-l3 detail** command.

Step 9 From an outside Linux client send ICMP ping to CSM vserver VIP-L3. This ping should be successful.

Step 10 On the active FWSM issue the **show xlate** command. You should see a global entry for each real in the serverfarm because only Layer 3 vservers load balance pings request to reals.

Step 11 Verify only vserver VIP-L3 has recorded policy match and packets received by issuing the following commands.

```
show module csm 2 vservers name DMZ1-FTP detail
show module csm 2 vservers name vip-dns detail
```

```
show module csm 2 vservers name vip-www detail
show module csm 2 vservers name vip-l3 detail
```

Step 12 Suspend the following vservers with the **no inservice** command: DMZ1-FTP, VIP-DNS, VIP-WWW, and VIP-L3.

Step 13 From an outside Linux client send ICMP ping to CSM vserver VIP-WWW. This ping should be unsuccessful because all four vserver configured with the same virtual IP have been taken out of service.

Step 14 Enable the following vservers with the **inservice** command.

```
Vsever DMZ1-FTP
VIP-DNS
VIP-WWW
VIP-L3
```

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the CSM not to load balance ICMP for a Layer 4 vserver.
- We expect the CSM to load balance ICMP for a Layer 3 vserver.
- We expect the FWSM to create a connection for ICMP when fixup protocol ICMP is configured.
- We expect vservers to respond to ICMP when operational.
- We expect a vserver not to respond to ICMP when not operational.

Results

ICMP to a CSM Layer 3 and Layer 4 Vserver (CSM Setup) passed.

Idle Timeout UDP (CSM Setup)

This test verified the CSM removed idle UDP connections at 60 seconds and the FWSM removed them after two minutes. It also verified that the CSM load balanced the UDP connections.

The CSM vserver VIP-TFTP has been configured with a 60 second idle timer. A TFTP copy request (UDP port 69) was generated on a Linux client, to the VIP-TFTP, in order to create a connection on the CSM and FWSM. It was verified that these connections were load balanced properly to the real servers in the serverfarm. It was then verified that these connections timed out after 60 seconds on the CSM and two minutes on the FWSM.

Relevant CSM Configuration:

```
!
 vserver VIP-TFTP
  virtual 101.40.40.244 udp 0
  vlan 301
  serverfarm FARM1-A
  advertise active
  idle 60
```

```
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the Idle Timeout UDP (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify CSM vserver VIP-TFTP is operational and the idle time out is set to 60 by using the **show mod csm 2 vserver name vip-tftp detail** command.

Step 3 Verify all reals are operational for CSM serverfarm FARM1-A by issuing the **show mod csm 2 real sfarm farm1-a detail** command.

Step 4 Clear all counters and connections on the CSM by issuing the **clear mod csm 2 counters** and **clear mod csm 2 conn** commands.

Step 5 On the Linux client dca-penguin-15, perform a single TFTP copy request to the VIP-TFTP using the **tftp 101.40.40.244 -c get file.txt** command.

Step 6 On the active CSM, use the **show mod csm 2 real sfarm farm1-a det** command to verify that UDP connections have been load balanced across the two real servers in serverfarm FARM1-A.

Each of the two real servers shows one connection apiece.

Step 7 On the active CSM, use the **show mod csm 2 conn vserver vip-tftp** command to verify that UDP connections have been created for the TFTP transfer.

Step 8 Use the **show clock** and **show mod csm 2 conn vserver vip-tftp** commands to verify that the UDP connections time out after one minute.

Step 9 Issue the **show timeout** command on the active FWSM in context VLAN 1101-2101 to verify timeout UDP is set to two minutes. In version 3.x the **show timeout** command does not exist. to view the timeouts do the **show run** command

Step 10 Issue the **clear conn** command on the active FWSM in context VLAN 1101-2101 to clear connections.

Step 11 On the Linux client dca-penguin-15, perform a single TFTP copy request to the VIP-TFTP using the **tftp 101.40.40.244 -c get file.txt** command.

Step 12 On the active FWSM, use the **show conn** command to verify that UDP connections have been created for the TFTP transfer.

Step 13 Use the **show clock** and **show conn** commands to verify that the UDP connections on the FWSM time out after two minutes.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect flows that exceed the idle timeout to be cleared from both the CSM and the FWSM.
- We expect the CSM vserver to properly load balance UDP traffic.

Results

Idle Timeout UDP (CSM Setup) passed.

Passive FTP Through FWSM and CSM (CSM Setup)

This test verified that the FWSM and CSM properly handled passive FTP traffic when the FTP fixup was enabled and disabled on the FWSM. FTP traffic was sent from outside client to vserver VIP-PASSIVE-FTP with FTP fixup enabled on the FWSM and when it was disabled. The same was done for FTP GET requests coming from an inside client to an outside server. In version 3.x of the FWSM code the ftp fixup protocol command has been superceeded by the Inspection engine. If you disable FTP inspection engines with the no inspect ftp command, outbound users can start connections only in passive mode, and all inbound FTP is disabled.

Relevant CSM Configuration:

```
!
module csm 2
 real P1_ETH1.3001
  address 101.1.1.11
  location dca-penguin-1
  inservice
 real P1_ETH1.3002
  address 101.1.1.12
  location dca-penguin-1
  inservice
!
 serverfarm FARM1-A
  nat server
  no nat client
  predictor leastconns
  real name P1_ETH1.3001
   inservice
  real name P1_ETH1.3002
   inservice
!
 vserver VIP-PASSIVE-FTP
  virtual 101.40.1.252 tcp ftp service ftp
  vlan 301
  serverfarm FARM1-A
  advertise active
  replicate csrp connection
  persistent rebalance
  inservice
```

!Relevant FWSM Configuration (context VLAN 1101-2101):

```
!
fixup protocol ftp
!
```

Test Procedure

The procedure used to perform the Passive FTP Through FWSM and CSM (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On dca-agg-1, using the **show module csm 2 vserver name vip-passive-ftp detail** command, verify that the CSM vserver VIP-PASSIVE-FTP is configured for service FTP and that it is pointing to the serverfarm FARM1-A.

The output of the command shows that the serverfarm that is being used is FARM1-A and that the **service = ftp** command is used.

Step 3 Using the **show module csm 2 serverfarm name farm1-a detail** command, verify that there are two real servers in serverfarm FARM1-A and that they are both operational.

Step 4 On the active FWSM, in context VLAN 1101-2101, use the **show fixup** command to verify that **fixup protocol ftp 21** is not configured. If it is configured, use the **no fixup protocol ftp** command to disable it. In version 3.x you need to execute the command **sh run policy-map** as the fixup is now done with the ftp inspection engine. If you can see inspect ftp, fixup is enabled.

Step 5 From an outside client, send a single passive FTP GET to vserver VIP-PASSIVE-FTP and verify that it fails.

The connection fails because the **fixup protocol ftp** has been disabled on the active FWSM.

Step 6 Send a single passive FTP request from an inside client to the outside server.

This connection should succeed. When FTP fixups have been disabled, only passive mode FTP is allowed from an inside interface to an outside interface (active FTP is disallowed).

Step 7 Configure **fixup protocol ftp 21** on the active FWSM context VLAN 1101-2101 to enable the fixup protocol for FTP on port 21.

Step 8 Issue the following commands on the active FWSM context VLAN 1101-2101 to clear connections and counters:

```
clear xlate
clear conn
clear logging buffer
```

Step 9 Issue the following commands on the active CSM to clear connections and counters:

```
clear module csm 2 counters
clear module csm 2 connections
```

Step 10 Send a single passive FTP GET request for a very large file from an outside client to the CSM vserver VIP-PASSIVE-FTP.

The target file, 100M_file is 100 megabytes in size.

Step 11 While the GET is under way, issue the following commands on the active FWSM context VLAN 1101-2101 to verify the FTP control and data channels were successfully created:

```
show conn
show xlate
show log
```

Step 12 While the GET is under way, issue the **show module csm 2 conn** command to verify the FTP control and data connections have been established.

Step 13 Send 20 passive FTP GETs from an outside client to the CSM vserver VIP-PASSIVE-FTP.

Each of these should succeed.

Step 14 On the active CSM, use the **show module csm 2 real sfarm farm1-a detail** command to verify that the previous GET requests have been load-balanced evenly across both servers in serverfarm FARM1-A.

Each real server listed in the output should show about the same number of total connections established.

Step 15 Send a single passive FTP request from inside the client to the outside server.

This connection should succeed.

Step 16 Stop background scripts to collect final status of network devices and analyze for error.

Step 17 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to permit passive FTP on the inside interface.
- We expect the FWSM to deny passive FTP on the outside interface when fixup protocol FTP 21 is disabled.
- If the FWSM code is 3.x then if you disable FTP inspection engines with the no inspect ftp command, outbound users can start connections only in passive mode, and all inbound FTP is disabled.
- We expect the CSM vserver to properly load balance active FTP.
- We expect no CPU or memory problems.

Results

Passive FTP Through FWSM and CSM (CSM Setup) passed.

CSM SSLM Focused

The tests in this section look at some of the interoperability capacities of the CSM and SSLSM, in terms of how they work together to handle data traffic.

This section contains the following topics:

URL Rewrite (CSM Setup)

This test verified that the SSLM properly manipulated the data coming from the server with the use of the URL rewrite functionality. Server data that contains a 300 series redirect will be rewritten to HTTPS being forwarded to the client.

HTTPS and HTTP traffic for this test is load balanced by a CSM.

IE, Firefox and a client emulator will be used to test basic SSL Termination and URL Rewrite

NOTE: Under the current time constraints we are not able to test every possible browser/version that exists today. The browsers were carefully selected to show any inconsistencies in SSL termination.

Relevant CSM configuration:

```
real DMZ1-SRV1
  address 192.168.100.100
  inservice
 real DMZ1-SRV2
```

```
  address 192.168.100.110
  inservice
!
 serverfarm DMZ1-CLEAR
  nat server
  no nat client
  real name DMZ1-SRV1 80
   inservice
  real name DMZ1-SRV2 80
   inservice
  probe TCP
!
 serverfarm SSLM-445
  nat server
  no nat client
  failaction reassign
  real name SSLM1 445
   inservice
  real name SSLM2 445
   inservice
  probe TCP
!
 sticky 200 ssl timeout 30
!
 vserver DMZ1-CLEAR
  virtual 172.16.100.200 tcp 445
  vlan 172
  serverfarm DMZ1-CLEAR
  persistent rebalance
  inservice
!
 vserver DMZ1-HTTPS
  virtual 192.168.100.200 tcp https
  serverfarm SSLM-445
  sticky 30 group 200
  persistent rebalance
  inservice Relevant SSLM1 configuration:

ssl-proxy policy url-rewrite dmz1-web
 url 192.168.100.200
 url 10.10.10.200
 url www.vip200.com
 url www.vip201.com
 url www.vip202.com
 url www.vip203.com
 url www.vip204.com
 url www.vip205.com
 url www.vip206.com
 url www.vip207.com
 url www.vip208.com
 url www.vip209.com
 url www.vip210.com
 url www.vip211.com
 url www.vip212.com
 url www.vip213.com
 url www.vip214.com
 url www.vip215.com
 url www.vip216.com
 url www.vip217.com
 url www.vip218.com
 url www.vip219.com
 url www.vip220.com
 url www.vip221.com
 url www.vip222.com
```

```
 url www.vip223.com
 url www.vip224.com
 url www.vip225.com
 url www.vip226.com
 url www.vip227.com
 url www.vip228.com
 url www.vip229.com
!
ssl-proxy service dmz1-web
 virtual ipaddr 172.16.100.100 protocol tcp port 445
 virtual policy ssl session-cache
 server ipaddr 172.16.100.200 protocol tcp port 445
 certificate rsa general-purpose trustpoint vip200
 inservice
```

!Relevant SSLM2 configuration:

```
ssl-proxy policy url-rewrite dmz1-web
 url 192.168.100.200
 url 10.10.10.200
 url www.vip200.com
 url www.vip201.com
 url www.vip202.com
 url www.vip203.com
 url www.vip204.com
 url www.vip205.com
 url www.vip206.com
 url www.vip207.com
 url www.vip208.com
 url www.vip209.com
 url www.vip210.com
 url www.vip211.com
 url www.vip212.com
 url www.vip213.com
 url www.vip214.com
 url www.vip215.com
 url www.vip216.com
 url www.vip217.com
 url www.vip218.com
 url www.vip219.com
 url www.vip220.com
 url www.vip221.com
 url www.vip222.com
 url www.vip223.com
 url www.vip224.com
 url www.vip225.com
 url www.vip226.com
 url www.vip227.com
 url www.vip228.com
 url www.vip229.com
!
ssl-proxy service dmz1-web
 virtual ipaddr 172.16.100.110 protocol tcp port 445
 virtual policy ssl session-cache
 server ipaddr 172.16.100.200 protocol tcp port 445
 certificate rsa general-purpose trustpoint vip200
 inservice
!
```

Test Procedure

The procedure used to perform the URL Rewrite (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Configure service rewrite-test with the url-rewrite policy rewrite-test by issuing the **policy url-rewrite rewrite-test** command on both SSL Modules.

Step 3 Verify url-rewrite policy url-rewrite has been successfully applied to service url-rewrite by issuing the **show ssl-proxy service rewrite-test** command on both SSL Modules.

Step 4 From the outside client use the client emulator to generate an HTTPS request to vserver SSL-REWRITE. Verify the location field of the HTTP 302 redirect packet was rewritten to HTTPS.

Step 5 Clear ssl-proxy service statistics and url statistics by issuing the following commands.

```
clear ssl-proxy stats service rewrite-test
clear ssl-proxy stats url
```

Step 6 Verify the ssl-proxy service statistics and url statistics have been cleared by issuing the following commands.

```
show ssl-proxy stats service rewrite-test
show ssl-proxy stats url
```

Step 7 Issue the **clear mod csm 5 count** command on the active CSM to clear csm counters.

Step 8 From the outside client use the client emulator to generate 1000 HTTPS request to vserver url-rewrite.

Step 9 When client emulated traffic has completed issue the **show ssl-proxy stats url** command on both SSLMs to verify the Rewrites Succeeded counter has incremented for a combined total of 1000.

Step 10 Issue the **show ssl-proxy stats service url-rewrite** command on both SSLMs to verify the conns attempted and full handshakes counters have incremented to 1000.

Step 11 On the Active CSM verify the total matches counter for vserver SSL-REWRITE and vserver CLEAR-REWRITE equals 2000 by issuing the command **show mod csm 5 vserver name***name***detail** command.

Step 12 On the Active CSM verify traffic was evenly load balanced between all reals in serverfarm SSLM-445 and serverfarm CLEAR-REWRITE by issuing the **show mod csm 2 real sfarm***name***detail** command.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the SSLM can rewrite the server issued 300 series redirects from HTTP to HTTPS.

Results

URL Rewrite (CSM Setup) passed.

Redundancy

The resiliency of network resources and services to hardware and software component failures is important to a successful high availability strategy in a data center network. These tests measure the effects of various failure scenarios on Layer 4-7 services and hardware.

This section contains the following topics:

CSM Redundancy Test (CSM Setup)

This test verified that flow information was replicate from the active CSM to the standby CSM. Upon a redundancy transition the standby CSM became the new active CSM and processed all flows that were originally created on the active CSM.

Test Procedure

The procedure used to perform the CSM Redundancy Test (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the **clear mod csm 2 counters** command to clear CSM counters on the active and standby CSM.

Step 3 Issue the following commands on the active and standby CSM to verify the counters have been cleared:

```
show mod csm 2 vservers name VIP-IXIA-SSLFE detail
show mod csm 2 vservers name VIP-IXIA-WWW detail
show mod csm 2 vservers name VIP-IXIA-FTP detail
show mod csm 2 vservers name VIP-IXIA-CLEAR detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm IXIAFARM-2 detail
show mod csm 2 real sfarm IXIAFARM-1 detail
show mod csm 2 real sfarm IXIAFARM-WWW detail
show mod csm 2 stats
show mod csm 2 conn
```

Step 4 Issue the **clear mod csm 2 sticky all** command on the active and standby CSM. Issue the **show mod csm 2 sticky** command to verify the sticky table was cleared.

Step 5 Issue the **clear ssl-proxy stats service** command on all SSLM's to clear statistics.

Step 6 Issue the **show ssl-proxy service** command on both SSLM's to verify all proxy services are operational.

Step 7 Generate HTTPS, HTTP and FTP traffic to vservers VIP-IXIA-SSLFE, VIP-IXIA-HTTP and VIP-IXIA-FTP.

Step 8 Issue the following commands on the active CSM to verify traffic flow, and to determine which reals connections are being stuck to:

```
show mod csm 2 vservers name VIP-IXIA-SSLFE detail
show mod csm 2 vservers name VIP-IXIA-WWW detail
```

```
show mod csm 2 vservers name VIP-IXIA-FTP detail
show mod csm 2 vservers name VIP-IXIA-CLEAR detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm IXIAFARM-2 detail
show mod csm 2 real sfarm IXIAFARM-1 detail
show mod csm 2 real sfarm IXIAFARM-WWW detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 9 Issue the following commands on the standby CSM to verify that connection information and sticky information has been replicated. Verify that the standby CSM is not load balancing any connections:

```
show mod csm 2 vservers name VIP-IXIA-SSLFE detail
show mod csm 2 vservers name VIP-IXIA-WWW detail
show mod csm 2 vservers name VIP-IXIA-FTP detail
show mod csm 2 vservers name VIP-IXIA-CLEAR detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm IXIAFARM-2 detail
show mod csm 2 real sfarm IXIAFARM-1 detail
show mod csm 2 real sfarm IXIAFARM-WWW detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 10 Issue the **show ssl-proxy stats service** command on all SSLSM's to verify the conns completed counter has incremented and that there are no handshake failures.

Step 11 Issue the **hw-module module 2 reset** command to reset the active CSM in slot 2.

Step 12 Issue the **show mod csm 2 ft** command on the standby to verify it is now the active CSM.

Step 13 Issue the following commands on the new active CSM to verify traffic flow and to determine if connections remained stuck to the same real:

```
show mod csm 2 vservers name VIP-IXIA-SSLFE detail
show mod csm 2 vservers name VIP-IXIA-WWW detail
show mod csm 2 vservers name VIP-IXIA-FTP detail
show mod csm 2 vservers name VIP-IXIA-CLEAR detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm IXIAFARM-2 detail
show mod csm 2 real sfarm IXIAFARM-1 detail
show mod csm 2 real sfarm IXIAFARM-WWW detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 14 Issue the **show ssl-proxy stats service** command to verify the conns completed counter has incremented and that there are no handshake failures.

Step 15 When the reloaded CSM comes back online, issue the **show mod csm 2 ft** command to verify it has preempted and is now the active CSM.

Step 16 Issue the following commands on the new active CSM to verify traffic flow and to determine if connections remained stuck to the same real:

```
show mod csm 2 vservers name VIP-IXIA-SSLFE detail
show mod csm 2 vservers name VIP-IXIA-WWW detail
show mod csm 2 vservers name VIP-IXIA-FTP detail
show mod csm 2 vservers name VIP-IXIA-CLEAR detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm IXIAFARM-2 detail
```

```
show mod csm 2 real sfarm IXIAFARM-1 detail
show mod csm 2 real sfarm IXIAFARM-WWW detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 17 Issue the following commands on the standby CSM to verify connection information and sticky information has been replicated:

```
show mod csm 2 vservers name VIP-IXIA-SSLFE detail
show mod csm 2 vservers name VIP-IXIA-WWW detail
show mod csm 2 vservers name VIP-IXIA-FTP detail
show mod csm 2 vservers name VIP-IXIA-CLEAR detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm IXIAFARM-2 detail
show mod csm 2 real sfarm IXIAFARM-1 detail
show mod csm 2 real sfarm IXIAFARM-WWW detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 18 Issue the **show ssl-proxy stats service** command to verify the conns completed counter has incremented and that there are no handshake failures.

Step 19 Stop background scripts to collect final status of network devices and analyze for error.

Step 20 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the CSM to replicate connections hitting the vserver.
- We expect the standby to become active and service the persistent replicated connection.
- We expect the CSM to preempt after a failure.
- We expect sticky connections to remain stuck to the same real after a failover.

Results

CSM Redundancy Test (CSM Setup) passed.

FWSM Redundancy (CSM Setup)

This test verified that long lived flows being load balanced by the CSM and traversing the FWSM will be replicated between the primary and secondary FWSM. The ability of the system to successfully replicate flows and forward traffic after the failover was the criterion for a successful test run.

Test Procedure

The procedure used to perform the FWSM Redundancy (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following commands on the primary and secondary FWSM:

```
show xlate      show conn
```

Step 3 Issue the **show failover** command on the primary and secondary FSWM to verify the primary FWSM is in active state.

Step 4 Use the **clear mod csm 2 count** command on the active CSM to clear counters.

Step 5 Issue the following commands to verify the counters have been cleared:

```
  show mod csm 2 vservers name VIP-IXIA-SSLFE detail  show mod csm 2 vservers name
VIP-IXIA-WWW detail  show mod csm 2 vservers name VIP-IXIA-FTP detail  show mod csm 2
vservers name VIP-IXIA-CLEAR detail  show mod csm 2 real sfarm SSLSM detail  show mod csm
2 real sfarm IXIAFARM-2 detail  show mod csm 2 real sfarm IXIAFARM-1 detail  show mod csm
2 real sfarm IXIAFARM-WWW detail  show mod csm 2 stats  show mod csm 2 conn
```

Step 6 Generate HTTPS traffic to vservers SSL30 and SSL29. Generate FTP traffic to vserver VIP1.

Step 7 Issue the following commands on the primary and secondary FWSM to verify connections:

```
show xlate      show conn
```

Step 8 Use the following commands on the active CSM to verify connections:

```
  show mod csm 2 vservers name VIP-IXIA-SSLFE detail  show mod csm 2 vservers name
VIP-IXIA-WWW detail  show mod csm 2 vservers name VIP-IXIA-FTP detail  show mod csm 2
vservers name VIP-IXIA-CLEAR detail  show mod csm 2 real sfarm SSLSM detail  show mod csm
2 real sfarm IXIAFARM-2 detail  show mod csm 2 real sfarm IXIAFARM-1 detail  show mod csm
2 real sfarm IXIAFARM-WWW detail  show mod csm 2 stats  show mod csm 2 conn
```

Step 9 Issue the **reload** command on the primary FWSM to force a reload.

Step 10 Issue the **show failover** command on the secondary FSWM to verify it is now active.

Step 11 Issue the following commands on the secondary FWSM to verify connections:

```
show xlate      show conn
```

Step 12 Issue the following commands on the active CSM several times to verify connections:

```
  show mod csm 2 vservers name VIP-IXIA-SSLFE detail  show mod csm 2 vservers name
VIP-IXIA-WWW detail  show mod csm 2 vservers name VIP-IXIA-FTP detail  show mod csm 2
vservers name VIP-IXIA-CLEAR detail  show mod csm 2 real sfarm SSLSM detail  show mod csm
2 real sfarm IXIAFARM-2 detail  show mod csm 2 real sfarm IXIAFARM-1 detail  show mod csm
2 real sfarm IXIAFARM-WWW detail  show mod csm 2 stats  show mod csm 2 conn
```

Step 13 When the primary FWSM comes back online, issue the **show failover** command to verify it is in standby state.

Step 14 Issue the following commands on the primary FWSM to verify connections have been replicated from the secondary FWSM:

```
show xlate      show conn
```

Step 15 Issue the **reload** command on the secondary FWSM to force a reload.

Step 16 Issue the **show failover** command on the primary FSWM to verify it is now active.

Step 17 Issue the following commands on the primary FWSM to verify connections:

```
show xlate      show conn
```

Step 18 Issue the following commands on the active CSM several times to verify connections:

```
  show mod csm 2 vservers name VIP-IXIA-SSLFE detail  show mod csm 2 vservers name
VIP-IXIA-WWW detail  show mod csm 2 vservers name VIP-IXIA-FTP detail  show mod csm 2
vservers name VIP-IXIA-CLEAR detail  show mod csm 2 real sfarm SSLSM detail  show mod csm
2 real sfarm IXIAFARM-2 detail  show mod csm 2 real sfarm IXIAFARM-1 detail  show mod csm
2 real sfarm IXIAFARM-WWW detail  show mod csm 2 stats  show mod csm 2 conn
```

Step 19 Wait for FTP traffic to complete and check for errors.

Step 20 Stop background scripts to collect final status of network devices and analyze for error.

Step 21 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to replicate flow information from active to standby FWSM.
- We expect the standby FWSM will transition to the active state with the failure of the active FWSM.

Results

FWSM Redundancy (CSM Setup) passed.

HSRP Failover (CSM Setup)

This test verified HSRP failover when a system failure occurred. This test also verified that the HSRP preempt command worked when the system returned to an operational state, if the interface was configured with a higher priority than the current active router interface in the same HSRP group. HTTPS traffic was sent through an FWSM and load balanced via CSM and SSLM.

Test Procedure

The procedure used to perform the HSRP Failover (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the **show standby brief** command on both Cisco 6500s to verify agg1 is active for all VLAN's.

Step 3 Issue the **clear mod csm 2 count** command on the active CSM to clear CSM counters. Issue the following commands to verify they have been cleared and to verify state: show mod csm 2 vservers name VIP-HOSTS-SSLBE detail show mod csm 2 vservers name VIP-HOSTS-SSL detail show mod csm 2 real sfarm SSLSM detail show mod csm 2 stats

Step 4 Issue the **show ssl-proxy service** command on all SSLSM's to verify the services are operational.

Step 5 Issue the **clear ssl-proxy stats service** command on all four SSLSM's to clear SSL-proxy service stats. Issue the **show ssl-proxy stats service** command to verify they have been cleared. Please note some counters might have incremented due to CSM probes.

Step 6 From outside client initiate HTTPS traffic to vserver VIP29 and VIP30.

Step 7 Issue the following commands on the active CSM to verify vservers SSL29, VIP29, SSL29, and VIP30 have open connections. show mod csm 2 vservers name VIP-HOSTS-SSLBE detail show mod csm 2 vservers name VIP-HOSTS-SSL detail show mod csm 2 real sfarm SSLSM detail show mod csm 2 stats

Step 8 Issue the following commands on all four SSLSM's to verify the conns attempted and conns completed counter has incremented and there are no errors.

- **show ssl-proxy stats service**
- **show ssl-proxy stats ssl**

Step 9 Issue the **reload** command on agg1 to force a failover.

Step 10 Issue the **show standby brief** command on agg2 to verify it is now active.

Step 11 Issue the following commands on the active CSM to verify vservers SSL29, VIP29, SSL29, and VIP30 have open connections.

Step 12 Issue the following commands on both SSLSM's in agg2 to verify the conns attempted and conns completed counter are still incrementing and there are no errors: show ssl-proxy stats service show ssl-proxy stats ssl

Step 13 When agg1 becomes operational again issue the **show standby brief** command to verify it preempts and becomes active.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the mean failover time for HSRP to be less than the default dead time of 10 seconds.
- We expect when the failed system becomes operational again, it will resume HSRP active status and forward traffic.

Results

HSRP Failover (CSM Setup) passed.

SSLM Reset (CSM Setup)

This test verified the effect of an SSL module reset on CSM load balancing. The CSM TCP probe detected the module failure and stopped load balancing traffic to it. The CSM continued to load balance traffic to the remaining operational SSL Module. When the CSM TCP probe detected the SSLM Module was operational again it started load balance traffic to it.

Relevant CSM Configuration:

```
real DMZ1-SRV1
  address 192.168.100.100
  inservice
real DMZ1-SRV2
  address 192.168.100.110
```

```
  inservice
real SSLM1
  address 172.16.100.100
  inservice
real SSLM2
  address 172.16.100.110
  inservice
!
 serverfarm DMZ1-CLEAR
  nat server
  no nat client
  real name DMZ1-SRV1 80
   inservice
  real name DMZ1-SRV2 80
   inservice
  probe TCP
!
  serverfarm SSLM-445
   nat server
   no nat client
   failaction purge
   real name SSLM1 445
    inservice
   real name SSLM2 445
    inservice
   probe TCP
!
 sticky 200 ssl timeout 30
!
vserver DMZ1-CLEAR
  virtual 172.16.100.200 tcp 445
  vlan 172
  serverfarm DMZ1-CLEAR
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver DMZ1-HTTPS
  virtual 192.168.100.200 tcp https
  serverfarm SSLM-445
  sticky 30 group 200
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
```

!Relevant SSLM1 Configuration:

```
ssl-proxy service dmz1-web
 virtual ipaddr 172.16.100.100 protocol tcp port 445
 virtual policy ssl session-cache
 server ipaddr 172.16.100.200 protocol tcp port 445
 certificate rsa general-purpose trustpoint vip200
 inservice
```

Relevant SSLM2 Configuration:

```
ssl-proxy service dmz1-web  virtual ipaddr
172.16.100.110 protocol tcp port 445
 virtual policy ssl session-cache
 server ipaddr 172.16.100.200 protocol tcp port 445
 certificate rsa general-purpose trustpoint vip200
 inservice
```

Test Procedure

The procedure used to perform the SSLM Reset (CSM Setup) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following to clear counters and stats on the active CSM

```
    clear mod csm 2 conn
    clear mod csm 2 counters
    clear mod csm 2 sticky allIssue the following commands to verify they have been
cleared:

    show mod csm 2 vservers name VIP-HOSTS-SSLBE detail
    show mod csm 2 vservers name VIP-HOSTS-BE detail
    show mod csm 2 real sfarm SSLSM detail
    show mod csm 2 real sfarm FARM1-BE detail
    show mod csm 2 serverfarms name SSLSM detail
    show mod csm 2 stats
    show mod csm 2 sticky
    show mod csm 2 conn
```

Step 3 Issue the following command on both SSLM's to verify the services are operational:

```
show ssl-proxy service
```

Step 4 Issue the following command on both SSLSM's to clear SSL-proxy service stats:

```
clear ssl-proxy stats service
```

Step 5 Issue the following command to verify they have been cleared:

```
show ssl-proxy stats service
```

Step 6 From an outside client initiate long lived HTTPS flow to vserver VIP30.

Step 7 Issue the following commands on the active CSM to verify vservers SSL29, SSL30, VIP30, and VIP29 have open connections.

```
    show mod csm 2 vservers name VIP-HOSTS-SSLBE detail
    show mod csm 2 vservers name VIP-HOSTS-BE detail
    show mod csm 2 real sfarm SSLSM detail
    show mod csm 2 real sfarm FARM1-BE detail
    show mod csm 2 serverfarms name SSLSM detail
    show mod csm 2 stats
    show mod csm 2 sticky
    show mod csm 2 conn
```

Step 8 Issue the following command on both SSLSM's to verify the conns attempted and conns completed counter has incremented and there are no errors:

```
show ssl-proxy stats service
```

Step 9 Issue the **hw-module module 3 reset** command on agg-1 reset SSLSM1.

Step 10 Monitor the client traffic. When the CSM probe detects a failure it should reset one of the active connections.

Step 11 When the CSM log message indicating the probe failure appears, send another HTTPS request from the client whose connections was reset.

Step 12 Issue the following commands on the active CSM to verify the TCP probe has failed real SSLM1 and all traffic is being load balanced to SSLM2:

```
show mod csm 2 vservers name VIP-HOSTS-SSLBE detail
show mod csm 2 vservers name VIP-HOSTS-BE detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-BE detail
show mod csm 2 serverfarms name SSLSM detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 13 Issue the following command to verify that the conns attempted and conns completed counter are still incrementing and there are no errors:

```
show ssl-proxy stats service
```

Step 14 After the SSLM becomes operational, generate multiple HTTPS request to vserver VIP-HOSTS-SSLBE.

Step 15 Issue the following commands to make sure traffic is being load balanced among the four SSLM's:

```
show mod csm 2 vservers name VIP-HOSTS-SSLBE detail
show mod csm 2 vservers name VIP-HOSTS-BE detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-BE detail
show mod csm 2 serverfarms name SSLSM detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 16 Stop background scripts to collect final status of network devices and analyze for error.

Step 17 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the CSM TCP probe to detect the SSLM failure.
- We expect the CSM to reset open connections when a probe fails a real.
- We expect the CSM to properly load balance traffic during a real failure.

Results

SSLM Reset (CSM Setup) passed.

CHAPTER 5

Layer 4-7 ACE

Tests in the "Layer 4-7 CSM" section on page 2-1 focus on traffic flows that touch multiple service modules. The service module combination that was used for this section was CSM+FWSM+SSLM+IDSM+NAM. Figure 1-1 illustrates the relevant topology for this section.

Refer to the following chapters for respective Layer 4-7 services testing:

- CSM—Based Integrated Switch Bundle "Layer 4-7 CSM" section on page 2-1
- CSM—Based Service Chassis Bundle "Layer 4-7 Services Switch" section on page 4-1
- Application Control Engine (ACE) "ACE" section on page 5-1
- Intrusion Detection Services Module (IDSM) "IDSM IPS" section on page 6-1

Test Results Summary

Table 5-1 on page 5-2 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 5-1 on page 5-2 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 5-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
ACE FWSM, page 5-3	n/a	1. Active FTP Through FWSM and ACE 2. DC Idle Timeout UDP 2 3. DNS Query Through ACE and FWSM 4. Passive FTP Through FWSM and ACE	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

ACE FWSM

The tests in this section look at some of the interoperability capacities of the ACE and FWSM, in terms of how they work together to handle data traffic.

This section contains the following topics:

Active FTP Through FWSM and ACE

This test verified that the FWSM and ACE properly handled active FTP traffic when the **ftp fixup protocol 21** and was enabled and disabled on the FWSM. FTP traffic was sent from an outside client to the ACE VIP and from an inside client to an outside server. In version 3.x of the FWSM code the ftp fixup protocol command has been superceeded by the Inspection engine. If you disable FTP inspection engines with the no inspect ftp command, outbound users can start connections only in passive mode, and all inbound FTP is disabled.

Relevant ACE context Configuration:

```
access-list anyone line 8 extended permit ip any any
rserver host SOURCE-NAT-11
  ip address 101.1.1.11
  inservice
rserver host SOURCE-NAT-12
  ip address 101.1.1.12
  inservice
serverfarm host NAT-SRVFARM
  rserver SOURCE-NAT-11
    inservice
  rserver SOURCE-NAT-12
    inservice
class-map match-all MATCH-SERVER-1
  2 match source-address 101.1.1.11 255.255.255.255
class-map match-all MATCH-SERVER-2
  2 match source-address 10.1.1.12 255.255.255.255
class-map match-all VIP-Address
  2 match virtual-address 101.3.1.201 tcp any
policy-map type loadbalance first-match VIP-LB
  class class-default
    serverfarm NAT-SRVFARM
policy-map multi-match SERVER-SRC-NAT
  class MATCH-SERVER-1
    nat static 101.3.1.200 netmask 255.255.255.255 vlan 3101
```

```
  class MATCH-SERVER-2
policy-map multi-match SERVER-VIP
  class VIP-Address
    loadbalance vip inservice
    loadbalance policy VIP-LB
    loadbalance vip icmp-reply
    loadbalance vip advertise
    inspect ftp
interface vlan 2101
  description Server side
  ip address 101.1.1.3 255.255.255.0
  alias 101.1.1.1 255.255.255.0
  peer ip address 101.1.1.2 255.255.255.0
  no normalization
  mac-sticky enable
  no icmp-guard
  access-group input anyone
  service-policy input SERVER-SRC-NAT
  no shutdown
interface vlan 3101
  description Client side
  ip address 101.3.1.101 255.255.255.0
  alias 101.3.1.102 255.255.255.0
  peer ip address 101.3.1.100 255.255.255.0
  no normalization
  no icmp-guard
  access-group input anyone
  service-policy input SERVER-VIP
  no shutdown
ip route 0.0.0.0 0.0.0.0 101.3.1.1
```

Test Procedure

The procedure used to perform the Active FTP Through FWSM and ACE test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following commands on the FWSM to clear connections and counters in the VLAN 1101-3101 context:

```
change context Vlan1101-3101
clear xlate
clear conn
clear access-list ACL-in count
clear logging buffer
```

Step 3 Issue the following commands on the FWSM to verify connections and counters have been cleared:

```
change context Vlan1101-3101
show xlate
show conn
show access-list ACL-in
```

Step 4 Issue the following commands on the Vlan1101-3101 context on the ACE:

```
clear service-policy SERVER-SRC-NAT
clear service-policy SERVER-VIP
clear conn
```

Step 5 Issue the following commands on the Vlan1101-3101 context on the ACE to verify the counters have been cleared:

```
show service-policy SERVER-VIP
show service-policy SERVER-SRC-NAT
show conn
```

Step 6 Send active FTP traffic to the FTP-context VIP from an outside client.

Step 7 Issue the following commands on the FWSM to verify the FTP control and data channels were successfully created:

```
change context Vlan1101-3101
show xlate
show conn
show log
```

Step 8 On the ACE issue the **show conn** and the **show service-policy SERVER-VIP**command to verify the FTP control and data connections have been established.

Step 9 When the FTP traffic has completed issue the following command on the FWSM to verify a match on the correct access list:

```
show access-list ACL-in | include extended permit tcp any 101.1.2.0 255.255.255.0 eq ftp
```

Step 10 Issue the following command on the Vlan1101-3101 context on the ACE to verify the FTP traffic was properly load balanced:

```
show rserver detail
show serverfarm detail
sh service-policy SERVER-VIP
```

Step 11 On the FWSM context VLAN 1101-3101, configure the **no fixup protocol ftp 21** command.

The **fixup protocol ftp 21** configuration is part of the default configuration for the DCAP test topology.

Step 12 Send an active FTP request from an inside client to an outside server.

This connection should fail. When the **no fixup protocol ftp 21** command has been configured only passive mode FTP is allowed from an inside interface to an outside interface.

Step 13 Issue the following commands on the FWSM to verify the FTP data channel was not successfully created:

```
change context Vlan1101-3101
show xlate
show conn
show log
```

Step 14 Reconfigure the **fixup protocol ftp 21** command on the VLAN 1101-3101 context to enable the fixup protocol for FTP on port 21 and use the **show fixup protocol ftp** command to verify it is now been enabled.

Step 15 Issue the following commands on the FWSM to clear connections and counters:

```
change context Vlan1101-2101
clear xlate
clear conn
clear access-list ACL-in count
```

Step 16 Send active FTP traffic to the FTP-context VIP from an outside client.

Step 17 Issue the following commands on the FWSM to verify the FTP control and data channels were successfully created:

```
change context Vlan1101-2101
show xlate
show conn
show log
```

Step 18 On the ACE issue the **show conn** and the **sh service-policy SERVER-VIP** command to verify the FTP control and data connections have been established.

Step 19 Stop background scripts to collect final status of network devices and analyze for error.

Step 20 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to permit active FTP on the outside interface.
- We expect the FWSM would deny active FTP on the inside to outside interface when fixup protocol FTP 21 is disabled in version 2.x
- If the FWSM code is 3.x then if you disable FTP inspection engines with the no inspect ftp command, outbound users can start connections only in passive mode, and all inbound FTP is disabled.
- We expect the ACE to properly load balance active FTP.

Results

Active FTP Through FWSM and ACE passed.

DC Idle Timeout UDP 2

This test verified the ACE removed idle UDP connections at 60 seconds and the FWSM removed them after two minutes. It also verified that the ACE load balanced the UDP connections.

The ACE context Vlan3101-2101 rserver has been configured with a 60 second idle timer. A TFTP copy request (UDP port 69) was generated on a Linux client, to the CLASS-VIP-1, in order to create a connection on the ACE and FWSM. It was verified that these connections were load balanced properly to the real servers in the serverfarm. It was then verified that these connections timed out after 60 seconds on the ACE and two minutes on the FWSM.

Relevant ACE context Vlan3101-2101 Configuration:

```
class-map match-all CLASS-VIP-1
  2 match virtual-address 101.3.1.200 udp any
parameter-map type connection IDLE_PARAMETER_MAP
  set timeout inactivity 60
policy-map multi-match Vlan3101-2101-POLICY
  class CLASS-VIP-1
    loadbalance vip inservice
```

```
loadbalance policy SRV-VIP-1
loadbalance vip icmp-reply
loadbalance vip advertiseconnection advanced-options IDLE_PARAMETER_MAPTest Procedure
```

The procedure used to perform the DC Idle Timeout UDP 2 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify ACE policy-map Vlan3101-2101-POLICY is inservice and the idle time out is set to 60 by using the **show service-policy Vlan3101-2101-POLICY** and the **sh parameter-map IDLE_PARAMETER_MAP** command.

Step 3 Verify all reals are operational for ACE serverfarm SRVFARM-1 by issuing the **show serverfarm SRVFARM-1** command.

Step 4 Clear all counters and connections on the ACE by issuing the **clear conn** and **clear service-policy Vlan3101-2101-POLICY** commands.

Step 5 On the Linux client dca-penguin-15, perform a single TFTP copy request to the VIP-TFTP using the **tftp 101.40.40.244 -c get file.txt** command.

Step 6 On the ACE context Vlan3101-2101, use the **show serverfarm SRVFARM-1 detail** command to verify that UDP connections have been load balanced across the two real servers in serverfarm SRVFARM-1 .

Each of the two real servers shows one connection apiece.

Step 7 On the ACE context Vlan3101-2101, use the **show conn** command to verify that UDP connections have been created for the TFTP transfer.

Step 8 Use the **show conn detail** commands to verify that the UDP connections time out after one minute.

Step 9 Issue the **show timeout** command on the active FWSM in context VLAN 1101-2101 to verify timeout UDP is set to two minutes. In version 3.x the **show timeout** command does not exist. to view the timeouts do the **show run** command

Step 10 Issue the **clear conn** command on the active FWSM in context VLAN 1101-3101 to clear connections.

Step 11 On the Linux client dca-penguin-15, perform a single TFTP copy request to the VIP-TFTP using the **tftp 101.40.40.244 -c get file.txt** command.

Step 12 On the active FWSM, use the **show conn** command to verify that UDP connections have been created for the TFTP transfer.

Step 13 Use the **show clock** and **show conn** commands to verify that the UDP connections on the FWSM time out after two minutes.

Step 14 On the ACE context Vlan3101-2101, use the **sh serverfarm SRVFARM-1 detail** command to verify that the previous GET requests have been load-balanced evenly across the servers in serverfarm SRVFARM-1.

Each real server listed in the output should show about the same number of total connections established.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect flows that exceed the idle timeout to be cleared from both the ACE and the FWSM.

- We expect the ACE service-policy to properly load balance UDP traffic.

Results

DC Idle Timeout UDP 2 passed.

DNS Query Through ACE and FWSM

This test verified that the FWSM and ACE properly handled DNS traffic when fixup protocol DNS was enabled.

Relevant Configuration on ACE context Vlan3101-2101:

```
class-map match-all CLASS-VIP-1
  2 match virtual-address 101.3.1.200 udp eq domainRelevant FWSM Access List:
```

access-list ACL-in extended permit udp any 201.1.1.0 255.255.255.0

Test Procedure

The procedure used to perform the DNS Query Through ACE and FWSM test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following commands on the active FWSM in context VLAN 1101-3101 to clear connections and counters:

- clear xlate
- clear conn
- clear access-list ACL-in count
- clear log buffer

Step 3 Issue the **clear service-policy Vlan3101-2101-POLICY** and **clear conn** commands on the ACE conttext Vlan3101-2101 to clear connections and counters.

Step 4 Use Spirent Avalanche or Ixia to send a small set of DNS queries to vserver VIP-DNS

Step 5 Issue the **show xlate** command on the active FWSM to verify that a global entry was created for the VIP in the ACE context Vlan3101-2101.

Step 6 Use the **show access-list ACL-in** command to verify there are matches on the portion of the access list that permits UDP DNS queries.

The ACL line that permits this traffic is:

access-list ACL-in extended permit udp any 201.1.1.0 255.255.255.0

Step 7 Use the following commands on the ACE context Vlan3101-2101 to verify the DNS traffic was properly load balanced:

- show service-policy Vlan3101-2101-POLICY
- show conn

The "total conns" should approximate the number of hits that was seen on the FWSM access-list.

Step 8 Use the **show serverfarm SRVFARM-1** command to verify that each real server in the serverfarm has made some connections.

The total conns established counter should be spread evenly accross the serverfarm

Step 9 Use the **clear service-policy Vlan3101-2101-POLICY** and **clear conn** commands on the ACE context to clear connections and counters.

Step 10 Use Spirent Avalanche or Ixia to send DNS queries to ACE context Vlan3101-2101 for domain name www.datacenter.com at rate of 1,000 users per second.

Step 11 While traffic is running issue the **show xlate** and **show conn | include most** commands on the VLAN 1101-3101 FWSM context to verify the xlate table and number of open connections.

Step 12 Verify the DNS connections rate on the Spirent Avalanche client stats tab and the select DNS. DNS queries per second should be around 1,000. Or on the Ixia stats screen check for the number of users and connection rate

Step 13 Use the following commands on the Ace context Vlan3101-2101 to verify the DNS traffic was properly load balanced. p>

```
show service-policy Vlan3101-2101-POLICY
show conn
sh serverfarm SRVFARM-1
```

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to permit DNS traffic to vserver DMZ2-DNS
- We expect the FWSM to NAT the DNS response to the outside client when fixup protocol DNS is enabled.
- We expect the FWSM not to NAT the DNS response to the inside client when fixup protocol DNS is enabled.
- We expect the ACE context Vlan3101-2101 to properly load balance DNS traffic.

Results

DNS Query Through ACE and FWSM passed.

Passive FTP Through FWSM and ACE

This test verified that the FWSM and ACE properly handled passive FTP traffic when the **ftp fixup protocol 21** and was enabled and disabled on the FWSM. FTP traffic was sent from an outside client to the ACE VIP and from an inside client to an outside server. In version 3.x of the FWSM code the ftp fixup protocol command has been superceeded by the Inspection engine. If you disable FTP inspection engines with the no inspect ftp command, outbound users can start connections only in passive mode, and all inbound FTP is disabled.

Relevant ACE context Configuration:

```
changeto Vlan3101-2101
configure terminal
```

```
class-map match-all CLASS-VIP-1
  2 match virtual-address 101.3.1.200 tcp eq ftp
end
```

Test Procedure

The procedure used to perform the Passive FTP Through FWSM and ACE test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On dca-ace1-agg-1 in the context Vlan3101-2101, using the **show service-policy Vlan3101-2101-POLICY detail** command, verify that the ACE Vlan3101-2101-POLICY is configured for service FTP and that it is pointing to the serverfarm SRVFARM-1.

The output of the command shows that the serverfarm that is being used is SRVFARM-1 and that the port equals the ftp port 21.

Step 3 Using the **show module csm 2 serverfarm name farm1-a detail** command, verify that there are two real servers in serverfarm FARM1-A and that they are both operational.

Step 4 On the active FWSM, in context VLAN 1101-3101, use the **show fixup** command to verify that **fixup protocol ftp 21** is not configured. If it is configured, use the **no fixup protocol ftp** command to disable it. In version 3.x you need to execute the command **sh run policy-map** as the fixup is now done with the ftp inspection engine. If you can see inspect ftp, fixup is enabled.

Step 5 From an outside client, send a single passive FTP GET to VIP CLASS-VIP-1 on context Vlan3101-2101 and verify that it fails.

The connection fails because the **fixup protocol ftp** has been disabled on the active FWSM.

Step 6 Send a single passive FTP request from an inside client to the outside server.

This connection should succeed. When FTP fixups have been disabled, only passive mode FTP is allowed from an inside interface to an outside interface (active FTP is disallowed).

Step 7 Configure **fixup protocol ftp 21** on the active FWSM context VLAN 1101-3101 to enable the fixup protocol for FTP on port 21.

Step 8 Issue the following commands on the active FWSM context VLAN 1101-3101 to clear connections and counters:

```
clear xlate
clear conn
clear logging buffer
```

Step 9 Issue the following commands on the ACE context Vlan3101-2101 to clear connections and counters:

```
clear conn
clear service-policy Vlan3101-2101-POLICY
```

Step 10 Send a single passive FTP GET request for a very large file from an outside client to the ACE context Vlan3101-2101.

The target file, 100M_file is 100 megabytes in size.

Step 11 While the GET is under way, issue the following commands on the active FWSM context VLAN 1101-2101 to verify the FTP control and data channels were successfully created:

```
show conn
```

```
show xlate
show log
```

Step 12 While the GET is under way, issue the **show conn** on the ACE context Vlan3101-2101 command to verify the FTP control and data connections have been established.

Step 13 Send 20 passive FTP GETs from an outside client to the CSM vserver VIP-PASSIVE-FTP.

Each of these should succeed.

Step 14 On the ACE context Vlan3101-2101, use the **sh service-policy Vlan3101-2101-POLICY** command to verify that the previous GET requests have been load-balanced evenly across the servers in serverfarm SRVFARM-1.

Each real server listed in the output should show about the same number of total connections established.

Step 15 Send a single passive FTP request from inside the client to the outside server.

This connection should succeed.

Step 16 Stop background scripts to collect final status of network devices and analyze for error.

Step 17 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to permit passive FTP on the inside interface.
- We expect the FWSM to deny passive FTP on the outside interface when fixup protocol FTP 21 is disabled.
- If the FWSM code is 3.x then if you disable FTP inspection engines with the no inspect ftp command, outbound users can start connections only in passive mode, and all inbound FTP is disabled.
- We expect the ACE context to properly load balance passive FTP.
- We expect no CPU or memory problems.

Results

Passive FTP Through FWSM and ACE passed.

CHAPTER 6

Layer 4-7 Services Switch

Tests in the "Layer 4-7 Services Switch" section on page 6-1 focus on the traffic flows that touch multiple service modules. The service module combination that was used for this section was CSM+FWSM+SSLM+IDSM+NAM in the service chassis model. Figure 1-3 illustrates the relevant topology for this section.

Refer to the following chapters for respective Layer 4-7 services testing:

- CSM—Based Integrated Switch Bundle "Layer 4-7 CSM" section on page 2-1
- ACE—Based Integrated Switch Bundle "Layer 4-7 ACE" section on page 3-1
- Application Control Engine (ACE) "ACE" section on page 5-1
- Intrusion Detection Services Module (IDSM) "IDSM IPS" section on page 6-1

Test Results Summary

Table 6-1 on page 6-2 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 6-1 on page 6-2 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 6-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
CSM FWSM, page 6-3	n/a	1. Active FTP Through FWSM and CSM Service Switch 2. DC Idle Timeout UDP Service Switch 3. DNS Query Through CSM and FWSM Service Switch 4. FWSM CSM Layer4 SYN Attack Service Switch 5. ICMP CSM L3 and L4 Vserver Service Switch 6. Passive FTP Through FWSM and CSM Service Switch	CSCsl39483
CSM SSLM Focused, page 6-12	n/a	1. Bundle SSL Sticky Service Switch 2. Bundled Backend SSL on Separate Service Switch 3. DC Cookie Sticky Spanning Packets 4. SSLM CIPHERS 5. URL Rewrite Service Switch	CSCec74017 CSCeh70549
Redundancy, page 6-22	n/a	1. Bundle HSRP Failover Service Switch 2. Bundle FWSM Redundancy Service Switch 3. Bundle SSLSM Reset Service Switch 4. CSM Redundancy Service Switch	CSCsj16292 CSCeh70549

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

CSM FWSM

The tests in this section look at some of the interoperability capacities of the CSM and FWSM, in terms of how they work together to handle data traffic.

This section contains the following topics:

Active FTP Through FWSM and CSM Service Switch

This test verified that the FWSM and CSM properly handled active FTP traffic when the **ftp fixup protocol 21** was enabled and disabled on the FWSM. FTP traffic was sent from an outside client to vserver VIP-ACTIVE-FTP and from an inside client to an outside server.

Test Procedure

The procedure used to perform the Active FTP Through FWSM and CSM Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following commands on the FWSM to clear connections and counters in the Vlan1101-2101 context:

- change context Vlan1101-2101
- clear xlate
- clear conn
- clear log
- conf t
- clear access-list ACL-in count

Step 3 Issue the following commands on the FWSM to verify connections and counters have been cleared:

- change context Vlan1101-2101
- show xlate
- show conn
- show access-list ACL-in

Step 4 Issue the following commands on the active CSM to clear connections and counters:

- clear mod csm 2 counters
- clear mod csm 2 connections

Step 5 Issue the following commands on the active CSM to verify the counters have been cleared:

- show mod csm 2 vserver name vip-active-ftp detail
- show mod csm 2 real sfarm farm1-b detail
- show mod csm 2 stats
- show mod csm 2 conns

Step 6 Send an active FTP request to vserver VIP-ACTIVE-FTP from an outside client.

Step 7 Issue the following command on the FWSM to verify the FTP control and data channels were successfully created:

- change context Vlan1101-2101
- show xlate
- show conn
- show log

Step 8 Issue the **show mod csm 2 conns** command to verify the FTP control and data connections have been established.

Step 9 When the FTP traffic has completed issue the following command on the FWSM to verify a match on the correct access list:

- show access-list ACL-in | include extended permit tcp any 201.1.1.0 255.255.255.0 eq ftp

Step 10 Issue the following command on the active CSM to verify the FTP traffic was properly load balanced:

- show mod csm 2 vserver name vip-active-ftp detail
- show mod csm 2 real sfarm farm1-b detail
- show mod csm 2 stats

Step 11 On the FWSM context Vlan1101-2101, configure the **no fixup protocol ftp 21** command.

The **fixup protocol ftp 21** command configuration is part of the default configuration for the DCAP test topology.

Step 12 Send an active FTP request from an inside client to an outside server.

This connection should fail. When the **no fixup protocol ftp 21** command has been configured, only passive mode FTP is allowed from an inside interface to an outside interface.

Step 13 Issue the following command on the FWSM to verify the FTP data channel was not successfully created:

- change context Vlan1101-2101
- show xlate
- show conn
- show log

Step 14 Reconfigure the **fixup protocol ftp 21** command on the Vlan1101-2101 context to enable the fixup protocol for FTP on port 21 and use the **show fixup protocol ftp** command to verify it is now been enabled.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to permit active FTP on the outside interface.
- We expect the FWSM would deny active FTP on the inside to outside interface when fixup protocol FTP 21 is disabled.
- We expect the CSM vserver to properly load balance active FTP.

Results

Active FTP Through FWSM and CSM Service Switch passed.

DC Idle Timeout UDP Service Switch

This test verified the CSM removed idle UDP connections at 60 seconds and the FWSM removed them after two minutes. It also verified that the CSM load-balanced the UDP connections.

The CSM vserver VIP-TFTP has been configured with a 60-second idle timer. A TFTP copy request (UDP port 69) was generated on a Linux client, to the VIP-TFTP, in order to create a connection on the CSM and FWSM. It was verified that these connections were load balanced properly to the real servers in the serverfarm. It was then verified that these connections timed out after 60 seconds on the CSM and two minutes on the FWSM.

Test Procedure

The procedure used to perform the DC Idle Timeout UDP Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify CSM vserver VIP-TFTP is operational and the idle timeout is set to 60 by issuing the **show mod csm 2 vserver name vip-tftp detail** command.

Step 3 Verify all reals are operational for CSM serverfarm FARM1-A by issuing the **show mod csm 2 real sfarm farm1-a detail** command.

Step 4 Clear all counters and connections on the CSM by issuing the **clear mod csm 2 counters** and **clear mod csm 2 conn** commands.

Step 5 On the Linux client dcb-penguin-11, perform a single TFTP copy request to the VIP-TFTP using the **tftp 201.40.40.244 -c get 100k_file.txt** command.

Step 6 On the active CSM, use the **show mod csm 2 serverfarm name farm1-a detail** command to verify that UDP connections have been created.

Step 7 On the active CSM, use the **show mod csm 2 conn vserver vip-tftp** command to verify that UDP connections have been created for the TFTP transfer.

Step 8 Use the **show clock** and **show mod csm 2 conn vserver vip-tftp** commands to verify that the UDP connections time out after one minute.

Step 9 Issue the **show running-config timeout** command on the active FWSM in context Vlan1101-2101 to verify timeout UDP is set to two minutes.

Step 10 Issue the **clear conn** command on the active FWSM in context Vlan1101-2101 to clear connections.

Step 11 On the Linux client dcb-penguin-11, perform a single TFTP copy request to the VIP-TFTP using the **tftp 201.40.40.244 -c get 100k_file.txt** command.

Step 12 On the active FWSM, use the **show conn | include UDP** command to verify that UDP connections have been created for the TFTP transfer.

Step 13 Use the **show clock** and **show conn | include UDP** commands to verify that the UDP connections on the FWSM timeout after two minutes.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect flows that exceed the idle timeout to be cleared from both the CSM and the FWSM.
- We expect the CSM vserver to properly load balance UDP traffic.

Results

DC Idle Timeout UDP Service Switch passed.

DNS Query Through CSM and FWSM Service Switch

This test verified that the FWSM and CSM properly handled DNS traffic when fixup protocol DNS was enabled. In this topology the CSM virtual is on the outside of the FWSM and the reals are on the inside of the FWSM. DNS requests to a farm of real servers running BIND were used to test the functionality of the CSM/FWSM combo.

Test Procedure

The procedure used to perform the DNS Query Through CSM and FWSM Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following commands on the active FWSM in context VLAN 1101-2101 to clear connections and counters:

- clear xlate
- clear conn
- clear access-list ACL-in count

- clear logg buffer

Step 3 In the FWSM Vlan1101-2101 context, use the **show running-config policy-map** command to verify that the fixup for DNS is configured.

Step 4 Use the **show module csm 2 vserver name vip-dns detail** command to verify that the CSM VIP is listening on UDP port 53, and that DNS queries are being sent to serverfarm FARM-DNS.

Step 5 Use the **show module csm 2 serverfarm name farm-dns detail** command to verify that there are 5 real servers in the DNS serverfarm and that they are all OPERATIONAL.

Step 6 Issue the **clear module csm 2 counters** and **clear module csm 2 connections** commands on the active CSM to clear connections and counters.

Step 7 Use the **host** command on dcb-penguin-11 to send a DNS query to vserver VIP-DNS for domain name dcb-penguin2.dcb-dcap.cisco.com.

Step 8 Issue the **show xlate** command on the active FWSM to verify that a global entry was created for each real in serverfarm FARM-DNS.

Step 9 Issue the **show access-list ACL-in | include udp any** command to verify there are matches on the portion of the access list that permits UDP DNS queries.

The ACL line that permits this traffic is:

access-list ACL-in extended permit udp any 201.1.1.0 255.255.255.0

Step 10 Issue the following commands on the active CSM to verify the DNS traffic was properly load balanced:

- show module csm 2 vserver name vip-dns detail
- show module csm 2 stats

The "total conns" should approximate the number of hits that was seen on the FWSM access-list.

Step 11 Issue the **host dcb-penguin1.dcb-dcap.cisco.com 201.40.40.247** command on dcb-penguin-11 ten more times and then issue the **show module csm 2 real sfarm farm-dns detail** command on dcb-ss-1 to verify that each real server in the serverfarm has made some connections.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to permit DNS traffic to vserver VIP-DNS.
- We expect the CSM vserver to properly load balance DNS traffic.

Results

DNS Query Through CSM and FWSM Service Switch passed.

FWSM CSM Layer4 SYN Attack Service Switch

SYN-flood attacks aim at preventing a TCP/IP server from servicing request. The SYN flag is set in a TCP segment when a connection request is sent by a computer. The target server responds back with an ACK and waits for a response from the initiator. The SYN-Flood attacker spoofs the source IP address so that the server never receives a response to the ACK. This causes the server to use up resources overloading the server and preventing it from responding to legitimate connection request.

TCP Intercept protects inside systems from a DoS attack perpetrated by flooding an interface with TCP SYN packets. Enable this feature by setting the maximum embryonic connections option of the NAT and static commands.

When the embryonic limit has been surpassed, the TCP intercept feature intercepts TCP SYN packets from clients to servers on a higher security level. The TCP intercept feature implements software to protect TCP servers from TCP SYN-flooding attacks, which are a type of DoS attack. SYN cookies are used during the validation process and help to minimize the amount of valid traffic being dropped.

The embryonic limit is a feature that is enabled for any inbound connection (a connection that the FWSM considers from lower to higher security). In order for a connection to be inbound, either hit a static or a global xlate.

This test verified the TCP Intercept feature by sending one million SYN packets generated on a Linux server using random source IP address. The SYN packets were sent to a CSM Layer 4 server with 65 reals behind the FWSM.

Test Procedure

The procedure used to perform the FWSM CSM Layer4 SYN Attack Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Configure context Vlan1101-2101 on the active FWSM with the following static command to enable the limitation of embryonic connections to 20:

```
static (inside,outside) 201.1.1.0 201.1.1.0 netmask 255.255.255.0 tcp 0 20
```

Step 3 Clear the translation table using the **clear xlate** command in the Vlan1101-2101 context.

Step 4 Issue the **clear module csm 2 counters** command to clear all CSM statistics and **clear module csm 2 conn** to clear all connections.

Step 5 Verify CSM utilization by issuing the **show module csm 2 tech-support utilization** command.

Step 6 On the FWSM system context, clear the Fast Path SYN Cookie Statistics Counters for NP-1 and NP-2 with the **clear np 1 syn** and **clear np 2 syn** commands in the system context.

Step 7 Verify CPU and memory utilization on the FWSM by issuing the **show cpu** and **show memory** commands from the system context.

Step 8 From the outside client send 10,000,000 SYN packets to vserver VIP-WWW with random source IP addresses.

Step 9 While the SYN attack traffic is being sent verify the rate of the SYN attack on the FWSM by issuing the **show perfmon | inc TCP Intercept** command. Issue the command several times to obtain a good baseline.

Step 10 While SYN attack traffic is being sent verify CSM utilization by issuing the **show module csm 2 tech-support utilization** command.

Step 11 Verify there are no errors on the CSM by issuing the following commands:

```
show mod csm 2 vserver name vip-www detail  show mod csm 2 reals sfarm farm1-a det  show mod csm 2 stats
```

Step 12 Verify the FWSM is issued a SYN cookie and verify the number of SYN packets intercepted by issuing the following commands:

```
show np 1 syn  show np 2 syn
```

Step 13 Verify FWSM CPU and memory utilization were not adversely impacted by the SYN attack by issuing the **show cpu** and **show memory** commands.

Step 14 Verify the FWSM log contains message number FWSM-6-106015 by issuing the **show log** command in context Vlan1101-2101.

Step 15 Remove static statement from VLAN 1101-2101 on the active FWSM with the following command:

```
no static (inside,outside) 201.1.1.0 201.1.1.0 netmask 255.255.255.0 tcp 0 20
```

Step 16 Stop background scripts to collect final status of network devices and analyze for error.

Step 17 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to intercept SYN packets being sent to the CSM reals by issuing a SYN cookie.
- We expect CPU and memory utilization on the CSM and FWSM not to be adversely impacted by the SYN attack.
- We expect the CSM to evenly load balance packets across all reals in the serverfarm.

Results

FWSM CSM Layer4 SYN Attack Service Switch failed. The following failures were noted: CSCsl39483.

ICMP CSM L3 and L4 Vserver Service Switch

This test verified ICMP ping traffic to multiple Layer 4 vservers and a Layer 3 vserver all configured with the same virtual IP address. The CSM virtual address was located on the outside of the FWSM and the CSM reals are located on the inside of the CSM.

Test Procedure

The procedure used to perform the ICMP CSM L3 and L4 Vserver Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the **clear module csm 2 counters** command to clear all CSM statistics.

Step 3 Issue the following commands on the active FWSM in context Vlan1101-2101 to clear connections and counters:

```
clear xlate
clear conn
clear logg buff
```

Step 4 Suspend CSM vserver VIP-L3 with the **no inservice** command.

Step 5 From an outside Linux client send an ICMP ping to CSM vserver VIP-WWW. This ping should be successful.

Step 6 On the active FWSM issue the **show xlate** command. You should see zero global entries because only Layer 3 vservers load balance pings to reals.

Step 7 Verify the following vservers have not recorded any policy matches or packets received by issuing the following commands:

show module csm 2 vservers name DMZ1-FTP detail

show module csm 2 vservers name vip-dns detail

show module csm 2 vservers name vip-www detail

show module csm 2 vservers name vip-l3 detail

Step 8 Enable CSM vserver VIP-L3 with the **inservice** command and verify it is now operational with the **show module csm 2 vserver vip-l3 detail** command.

Step 9 From an outside Linux client send ICMP ping to CSM vserver VIP-L3. This ping should be successful.

Step 10 On the active FWSM issue the **show xlate** command. You should see a global entry for each real in the serverfarm because only Layer 3 vservers load balance pings request to reals.

Step 11 Verify only vserver VIP-L3 has recorded policy match and packets received by issuing the following commands:

```
show module csm 2 vservers name DMZ1-FTP detail
show module csm 2 vservers name vip-dns detail
show module csm 2 vservers name vip-www detail
show module csm 2 vservers name vip-l3 detail
```

Step 12 Suspend the following vservers with the **no inservice** command: DMZ1-FTP, VIP-DNS, VIP-WWW, and VIP-L3.

Step 13 From an outside Linux client send ICMP ping to CSM vserver VIP-WWW. This ping should be unsuccessful because all four vserver configured with the same virtual IP have been taken out of service.

Step 14 Enable all of the vservers with the **inservice** command.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the CSM not to load balance ICMP for a Layer 4 vserver.
- We expect the CSM to load balance ICMP for a Layer 3 vserver.

- We expect the FWSM to create a connection for ICMP when fixup protocol ICMP is configured.
- We expect vservers to respond to ICMP when operational.
- We expect a vserver not to respond to ICMP when not operational.

Results

ICMP CSM L3 and L4 Vserver Service Switch passed.

Passive FTP Through FWSM and CSM Service Switch

This test verified that the FWSM and CSM properly handled passive FTP traffic when the FTP fixup was enabled and disabled on the FWSM. FTP traffic was sent from an outside client to vserver VIP-PASSIVE-FTP with FTP fixup enabled on the FWSM and when it was disabled. The same was done for FTP GET requests coming from an inside client to an outside server.

Test Procedure

The procedure used to perform the Passive FTP Through FWSM and CSM Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On dcb-ss-1, using the **show module csm 2 vserver name vip-passive-ftp detail** command, verify that the CSM vserver VIP-PASSIVE-FTP is configured for service FTP and that it is pointing to the serverfarm FARM1-B.

The output of the command shows that the serverfarm that is being used is FARM1-B and that **service = ftp** command is enabled.

Step 3 Using the **show module csm 2 serverfarm name farm1-b detail** command, verify that there are two real servers in serverfarm FARM1-B and that they are both operational.

Step 4 On the active FWSM, in context Vlan1101-2101, use the **show fixup** command to verify that fixup protocol FTP 21 is not configured. If it is configured, use the **no fixup protocol ftp** command to disable it. In version 3.x you need to execute the command **sh run policy-map** as the fixup is now done with the ftp inspection engine. If you can see inspect ftp, fixup is enabled.

Step 5 From an outside client, send a single passive FTP GET to vserver VIP-PASSIVE-FTP and verify that it fails.

The connection fails because the **fixup protocol ftp** has been disabled on the active FWSM.

Step 6 Send a single passive FTP request from an inside client to the outside server.

This connection should succeed. When FTP fixups have been disabled, only passive mode FTP is allowed from an inside interface to an outside interface (active FTP is disallowed).

Step 7 Configure **fixup protocol ftp 21** on the active FWSM context Vlan1101-2101 to enable the fixup protocol for FTP on port 21.

Step 8 Issue the following commands on the active FWSM context Vlan1101-2101 to clear connections and counters:

- clear xlate
- clear conn
- clear log

Step 9 Issue the following commands on the active CSM to clear connections and counters:

```
clear module csm 2 counters
clear module csm 2 connections
```

Step 10 Send a single passive FTP GET request for a very large file from an outside client to the CSM vserver VIP-PASSIVE-FTP.

The target file, 1G_file.zip is 1-Gigabyte in size.

Step 11 While the GET is under way, issue the following commands on the active FWSM context VLAN 1101-2101 to verify the FTP control and data channels were successfully created:

```
show conn
show xlate
show log
```

Step 12 While the GET is under way, issue the **show module csm 2 conn** command to verify the FTP control and data connections have been established.

Step 13 Send 20 passive FTP GET's from an outside client to the CSM vserver VIP-PASSIVE-FTP.

Each of these should succeed.

Step 14 On the active CSM, use the **show module csm 2 real sfarm farm1-b detail** to verify that the previous GET requests have been load balanced evenly across both servers in serverfarm FARM1-B.

Each real server listed in output should show about the same number of total connections established.

Step 15 Send a single passive FTP request from an inside client to the outside server.

This connection should succeed.

Step 16 Stop background scripts to collect final status of network devices and analyze for error.

Step 17 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to permit passive FTP on the inside interface.
- We expect the FWSM to deny passive FTP on the outside interface when fixup protocol FTP 21 is disabled.
- We expect the CSM vserver to properly load balance active FTP.
- We expect no CPU or memory problems.

Results

Passive FTP Through FWSM and CSM Service Switch passed.

CSM SSLM Focused

The tests in this section look at some of the interoperability capacities of the CSM and SSLSM, in terms of how they work together to handle data traffic.

This section contains the following topics:

Bundle SSL Sticky Service Switch

This test verified the ability of the CSM to extract SSL Session ID and add an SSL entry to the sticky table. Subsequent SSL requests containing the same SSL Session ID were sent to the same real server associated with that sticky entry. The real servers used in this test were SSL modules.

Test Procedure

The procedure used to perform the Bundle SSL Sticky Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the **clear mod csm 2 counters** command on the active CSM. Issue the following commands to verify the counters have been cleared:

```
show mod csm 2 vservers name SSL30 detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 stats
```

Step 3 Issue the **clear mod csm 2 sticky all** command on the active CSM to clear all sticky entries.

Step 4 Issue the **show mod csm 2 sticky** command to verify all SSL sticky entries have been cleared.

Step 5 Issue the **show ssl-proxy service** command on all four SSLMs to verify SSL-proxy service is operational.

Step 6 Issue the **clear ssl-proxy stats service** command on all four SSLMs to clear SSL-proxy service statistics.

Step 7 Issue the **show ssl-proxy stats service** command on all four SSLMs to verify statistics have been cleared.

Step 8 Begin initiating SSL GET requests to vserver SSL30. This involves a single user generating 240 HTTPS requests where a new SSL Session ID will be generated on every 30th request.

Step 9 Within a few seconds after the traffic has started, issue the **show module csm 2 reals sfarm sslsm detail** command on the active CSM to verify that all of the connections up to this point are being sent ("stuck") to a single SSLSM server.

The **total connections established** on one of the servers should be some value greater than 1 and less than 30. There should be no established connections on any of the other servers.

Step 10 When traffic has completed, verify that connections were load balanced among the four SSLMs in serverfarm SSLMSM:

```
show mod csm 2 vservers name SSL30 detail
show mod csm 2 real sfarm SSLSM detail
```

```
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 stats
```

Step 11 Use the **show module csm 2 sticky group 206** command on the active CSM to verify that the SSL sticky group has entries in it.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the CSM to stick clients to the same real based on SSL Session ID.

Results

Bundle SSL Sticky Service Switch passed.

Bundled Backend SSL on Separate Service Switch

This test verified that the CSM and SSLM successfully worked together to load balance SSL traffic on the client side, internally decrypt the traffic for advanced Layer 7 processing then re-encrypt the traffic load balancing to the backend servers. This test also verified the CSM was able to stick clients to the same real based on SSL ID.

The CSM and SSLM communicate together on an internal VLAN in routed mode. The CSM communicates with the clients and reals in bridged mode. Clients access the CSM virtual addresses through static NAT mappings on the FWSM.

Test Procedure

The procedure used to perform the Bundled Backend SSL on Separate Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following command on all four SSLMs to clear statistics:

```
clear ssl-proxy stats hdr
clear ssl-proxy stats ssl
clear ssl-proxy stats service
```

Step 3 Issue the following commands on all four SSLMs to verify statistics have been cleared:

```
show ssl-proxy stats service
show ssl-proxy stats hdr
show ssl-proxy stats ssl client
```

Step 4 Issue the **show ssl-proxy service** command on all four SSLSM's to verify SSL-proxy services are operational.

Step 5 Issue the **clear mod csm 2 counters** command on the active CSM to clear counters.

Step 6 Issue the following commands to verify the counters have been cleared:

```
show mod csm 2 vserver name SSL30 detail
show mod csm 2 vserver name VIP30 detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 stats
```

Step 7 Issue the **clear mod csm 2 sticky all** command on the active CSM to clear the sticky table.

Step 8 Issue the **show mod csm 2 sticky** command on the active CSM to verify the sticky table has been cleared.

Step 9 Send multiple HTTPS GET requests for 1.html, 2.html, and 3.html from the outside client to vserver SSL30. The client emulation tool will generate the traffic using three different cookies.

The test tool is configured to send 1x requests for the 1.html files, 2x requests for the 2.html files, and 3x requests for the 3.html files.

Step 10 Wait until client emulation traffic has completed, then issue the **show mod csm 2 vservers name ssl30 detail** command to verify the **Tot matches** counter equals 1200.

Step 11 Issue the **show mod csm 2 vservers name vip30 detail** command to verify the **Tot matches** counter has incremented for the following three policies:

```
200 times for 1.HTML
400 times for 2.HTML
600 times for (default)
```

Step 12 Issue the **show mod csm 2 real sfarm farm1-b-be detail** command on the active CSM to verify the load balancing of connections.

Step 13 Issue the **show mod csm 2 stats** command on the active CSM to verify there are no errors.

Step 14 Issue the **show mod csm 2 sticky** command on the active CSM to verify the sticky table.

Step 15 Issue the **show ssl-proxy stats service backend** command on all SSLMs to verify the following two counters equal 720:

```
conns attempted
conns completed
```

Step 16 Issue the **show ssl-proxy stats service BACKENDCLIENT** command on all four SSLMs to verify that the conns attempted and conns completed counters have incremented and there are no errors.

Step 17 Stop background scripts to collect final status of network devices and analyze for error.

Step 18 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the CSM to correctly load balance SSL traffic.
- We expect the CSM to apply the correct Layer 7 policy on clear text traffic.
- We expect the CSM to be able to stick based on the SSL ID.
- We expect the SSLSM to reencrypt the clear text traffic and forward through the CSM to the backend server.

- We expect the SSLSM to insert client IP and port information.
- We expect the SSLM to insert the customer header.

Results

Bundled Backend SSL on Separate Service Switch passed.

DC Cookie Sticky Spanning Packets

This test verified that the CSM properly inserted a cookie and provided persistance based on the cookie sent in the http client request while the MTU on the client was set to 576.

Internet Explorer and Firefox, were used to test cookie insert and cookie persistance.

NOTE: Under the current time constraints we are not able to test every possible browser/version that exists.

Relevant CSM Configuration:

```
!
 vserver WWWIN-OEFIN
  virtual 201.40.30.51 any
  vlan 301
  serverfarm ORACLE-ALL
  advertise active
  sticky 30 group 2
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  domain dcb-csm-1
  inservice
!
 vserver WWWIN-REDIRECT
  virtual 201.40.30.51 tcp www
  serverfarm 80-TO-8000
  persistent rebalance
  inservice
!
dcb-ss-1#show mod csm 2 sticky config
Group  CurrConns Timeout  Type
------------------------------------------------------------
1      0         60       src-ip netmask 255.255.255.255
2      2         30       cookie-insert CHOCO
3      0         30       cookie backend-ssl
10     0         30       src-ip netmask 255.255.255.255
30     0         30       src-ip netmask 255.255.255.255
99     0         1        both-ip netmask 255.255.255.0
!
dcb-ss-1#show mod csm 2 sticky group 2
group   sticky-data              real                  timeout
----------------------------------------------------------------
2       cookie D5CB4378:8D42CD15 201.1.33.5            0
2       cookie ED20F247:C3EB50D3 201.1.33.16           0
!
```

Test Procedure

The procedure used to perform the DC Cookie Sticky Spanning Packets test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 From the outside client, set the MTU (Maximum Transmission Unit) to 576 bytes in order to simulate a dial-up or satellite user.

Step 3 Clear the counters on the CSM by issuing the **clear mod csm 2 counters** command.

Step 4 Clear the sticky table on the CSM by issuing the **clear mod csm 2 sticky all** command.

Step 5 Send some HTTP GET requests from Internet Explorer to http://wwwin-oefin.gslb.dcap.com:8000/index.html Verify on the CSM the real that serviced this request, by issuing the **show mod csm 2 reals sfarm oracle-all detail** command.

Step 6 Send some HTTP GET reqeusts from Firefox version 2.x browser to http://wwwin-oefin.gslb.dcap.com:8000/index.html Verify on the CSM the real that serviced this request by issuing the **show mod csm 2 reals sfarm oracle-all detail**.

Step 7 Stop the packet capture. Parse the output to verify the CSM inserted a unique cookie for each browser used.

Step 8 Start a packet capture for the client traffic.

Step 9 Hit refresh several times on the IE x.0 browser. Verify on the CSM the same real as before, by issuing the **show mod csm 2 reals sfarm oracle-all detail** command.

Step 10 Hit refresh several times on the Firefox 2.x browser. Verify on the CSM the same real as before, by issuing the show mod csm 2 reals sfarm oracle-all detail command.

Step 11 On the client, set the MTU back to its original setting.

Step 12 Send a POST requests to wwwin-oefin.gslb.dcap.com by sending and HTTP POST on the E-Business Link located at the following URL: http://wwwin-oefin.gslb.dcap.com:8000/OA_HTML/AppsLocalLogin.jsp?requestUrl=APPSHOMEPAGE&cancelUrl=http%3A%2F%2Fwwwin-oefin.gslb.dcap.com%3A8000%2Foa_servlets%2Foracle.apps.fnd.sso.AppsLogin

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the CSM to insert an HTTP cookie into the http data stream via an http cookie header. We expect the CSM to provide persistance for client connections when the client presents the CSM with the cookie.

Results

DC Cookie Sticky Spanning Packets passed.

SSLM CIPHERS

This test verified that the SSLM properly manipulated the data coming from the server with the use of the URL rewrite functionality. Server data that contains a 300 series redirect will be rewritten to HTTPS being forwarded to the client.

HTTPS and HTTP traffic for this test is load balanced by a CSM.

Internet Explorer, Firefox, and a client emulator will be used to test basic SSL Termination and URL Rewrite.

NOTE: Under the current time constraints we are not able to test every possible browser/version that exists. The browsers were carefully selected to show any inconsistencies in SSL termination.

Relevant CSM Configuration:

```
!
 sticky 241 ssl timeout 30
!
 vserver CLEAR-REWRITE
  virtual 201.40.40.241 tcp www
  vlan 302
  serverfarm FARM1-A
  persistent rebalance
  inservice
!
 vserver SSL-REWRITE
  virtual 201.40.40.241 tcp https
  vlan 301
  serverfarm SSLSM
  advertise active
  sticky 30 group 241
  persistent rebalance
  inserviceRelevant SSLM1 Configuration:

!
 policy url-rewrite rewrite-test
  url 201.40.40.241
  url www.urlrewrite.com
  url www.urlrewrite1.com
  url www.urlrewrite2.com
  url www.urlrewrite3.com
  url www.urlrewrite4.com
  url www.urlrewrite5.com
  url www.urlrewrite6.com
  url www.urlrewrite7.com
  url www.urlrewrite8.com
  url www.urlrewrite9.com
  url www.urlrewrite10.com
  url www.urlrewrite11.com
  url www.urlrewrite12.com
  url www.urlrewrite13.com
  url www.urlrewrite14.com
  url www.urlrewrite15.com
  url www.urlrewrite16.com
  url www.urlrewrite17.com
  url www.urlrewrite18.com
  url www.urlrewrite19.com
  url www.urlrewrite20.com
  url www.urlrewrite21.com
  url www.urlrewrite22.com
  url www.urlrewrite23.com
```

```
 url www.urlrewrite24.com
 url www.urlrewrite25.com
 url www.urlrewrite26.com
 url www.urlrewrite27.com
 url www.urlrewrite28.com
 url www.urlrewrite29.com
 url www.urlrewrite30.com
!
service url-rewrite
 virtual ipaddr 201.40.40.241 protocol tcp port 443 secondary
 server ipaddr 200.1.0.20 protocol tcp port 80
 certificate rsa general-purpose trustpoint url-rewrite
 no nat server
 policy url-rewrite rewrite-test
 inservice
!
```

Test Procedure

The procedure used to perform the SSLM CIPHERS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the Client machine, Branch3-client-1.cisco.com set the MTU to 576

Step 3 Clear SSL-proxy service statistics by using the following commands:

```
clear ssl-proxy stats service rewrite-test
```

Step 4 Verify the SSL-proxy service statistics have been cleared on the SSLM in the topology by using the following commands:

```
show ssl-proxy stats service rewrite-test
show ssl-proxy stats service rewrite-test detail
```

Step 5 Issue the **clear mod csm 2 count, clear mod csm 2 conn, and clear mod csm sticky all** command, on the active CSM to clear CSM stats.

Step 6 Startup Wireshark or Ethereal on the client **branch3-client-1** and make sure it is listening on TCP port 443.

Step 7 Verify the ssl ciphers that are configured on the SSLM for ssl-proxy service "rewrite-test" are set to **rsa-with-rc4-128-md5** and that **version all** is selected.

Step 8 On Brnach-client-1, verify that the IE browser is configured to send both SSL 3.0 and TLS 1.0 record layer/client hello packets.

Step 9 From the Branch2-client-1 client, with ethereal or wireshark listening on tcp port 443, Open Internet Explorer and initiate a connection to **https://201.40.40.241/3.html**

Step 10 Stop the ethereal or wireshark packet trace and check the client/server hello for cipher suites offered and accepted and check for SSL record layer TLS 1.0. Verify the SSL record layer version sent by the browser is TLS 1.0. Verify the correct cipher was chosen by the SSLM by looking in the Server Hello. Verify there are no encrypted alerts and note the ciphers sent by the browser.

Step 11 Verify the symmetric operations and cipher suites chosen by the SSLM by issuing the **"show ssl-proxy stats service rewrite-test detail"** command.

Step 12 Modify the ssl ciphers that are configured on the SSLM for ssl-proxy service "rewrite-test" to include only the cipher **rsa-with-rc4-128-sha** and that **version all** is selected.

Step 13 Startup Wireshark or Ethereal again on the client **branch2-client-1** and make sure it is listening on TCP port 443.

Step 14 From the Branch2-client-1 client, with ethereal or wireshark listening on tcp port 443, Open Internet Explorer and initiate a connection to **https://201.40.40.241/3.html**

Step 15 Stop the ethereal or wireshark packet trace and check the client/server hello for cipher suites offered and accepted and check for SSL record layer TLS 1.0. Verify the SSL record layer version sent by the browser is TLS 1.0. Verify the correct cipher was chosen by the SSLM by looking in the Server Hello. Verify there are no encrypted alerts and note the ciphers sent by the browser.

Step 16 Verify the symmetric operations and cipher suites chosen by the SSLM by issuing the **"show ssl-proxy stats service rewrite-test detail"** command.

Step 17 Modify the ssl ciphers that are configured on the SSLM for ssl-proxy service "rewrite-test" to include only the cipher **rsa-with-des-cbc-sha** and that **version all** is selected.

Step 18 Verify the SSL-proxy service statistics have been cleared on the SSLM in the topology by using the following commands:

```
clear ssl-proxy stats service rewrite-test
clear ssl-proxy session service rewrite-test
clear ssl-proxy conn service rewrite-test
```

Step 19 From the Branch3-client-1 client, with ethereal or wireshark listening on tcp port 443, Open Internet Explorer and initiate a connection to **https://dcap-frontend**

Step 20 Verify the symmetric operations and cipher suites chosen by the SSLM by issuing the **"show ssl-proxy stats service rewrite-test detail"** command.

Step 21 Modify the ssl ciphers that are configured on the SSLM for ssl-proxy service "rewrite-test" to include only the cipher **rsa-with-3des-ede-cbc-sha** and that **version all** is selected.

Step 22 Verify the SSL-proxy service statistics have been cleared on the SSLM in the topology by using the following commands:

```
clear ssl-proxy stats service rewrite-test
clear ssl-proxy session service rewrite-test
clear ssl-proxy conn service rewrite-test
```

Step 23 From the Branch3-client-1 client, with ethereal or wireshark listening on tcp port 443, Open Internet Explorer and initiate a connection to **https://201.40.40.241/3.html**

Step 24 Verify the symmetric operations and cipher suites chosen by the SSLM by issuing the **"show ssl-proxy stats service rewrite-test detail"** command.

Step 25 Stop background scripts to collect final status of network devices and analyze for error.

Step 26 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the SSLM will terminate SSL connections using various ciphers.
- We expect that the SSLM will terminate SSL traffic without error.

- We expect that the SSLM selects the correct cipher based on the configured ciphers in the ssl proxy service.

Results

SSLM CIPHERS passed.

URL Rewrite Service Switch

This test verified that the SSLM properly manipulated the data coming from the server with the use of the URL rewrite functionality. Server data that contains a 300 series redirect will be rewritten to HTTPS being forwarded to the client.

HTTPS and HTTP traffic for this test is load balanced by a CSM. IE, Firefox and a client emulator test tool will be used to test basic SSL Termination and URL Rewrite

NOTE: Under the current time constraints we are not able to test every possible browser/version that exists today. The browsers were carefully selected to show any inconsistencies in SSL termination.

Test Procedure

The procedure used to perform the URL Rewrite Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Use the **show module csm 2 vserver name ssl-rewrite detail** command to verify that the vserver is operational and sending incoming connections to serverfarm SSLSM.

Step 3 Use the **show module csm 2 vserver name clear-rewrite detail** command to verify that the vserver is operational and sending incoming connections to serverfarm CLEAR-REWRITE.

Step 4 Verify that the url-rewrite policy rewrite-test has been configured to match the URL string 201.40.40.241 by issuing the **show ssl-proxy policy url-rewrite rewrite-test** command on all four SSL Modules.

Step 5 Verify that the SSL-proxy service rewrite-test is configured and operational with the url-rewrite policy rewrite-test by issuing the **show ssl-proxy service rewrite-test** command on all four SSL modules.

The Operation Status on each SSLM should be "up".

Step 6 Use the client emulator test tool to generate a single HTTPS request to vserver SSL-REWRITE. Verify the location field of the HTTP 302 redirect packet was rewritten to HTTPS.

The server should reply with a HTTP return code of 302 and a redirect location of https://201.40.40.241/2.gif. The SSL Stats summary should show a single ssl_redirect having taken place, and two no_redirects.

Step 7 Clear ssl-proxy service statistics and url statistics by issuing the following commands:

```
clear ssl-proxy stats service rewrite-test
clear ssl-proxy stats url
```

Step 8 Verify the ssl-proxy service statistics and url statistics have been cleared by issuing the **show ssl-proxy stats service rewrite-test** and **show ssl-proxy stats url** commands.

Note that for the valid sessions counter to clear, one has to do "clear ssl-proxy session" which will "clear" the active sessions count also.

Step 9 Issue the **clear module csm 2 counters** command on the active CSM to clear the CSM counters.

Step 10 Use the client emulator to generate 1000 HTTPS requests to vserver url-rewrite.

Step 11 When client emulated traffic has completed issue the **show ssl-proxy stats url** command on all four SSLMs to verify the Rewrites Succeeded counter has incremented for a combined total of 1000.

Step 12 Issue the **show ssl-proxy stats service rewrite-test** command on all four SSLMs to verify the conns attempted counters have incremented to a total of 2000. Verify the same for the full handshakes and conns completed counters.

Though 1000 client requests were sent, the SSLMs will show a total of 2000 connections because of the initial request and the redirected request.

Step 13 On the active CSM, verify the Tot matches counter for vservers SSL-REWRITE and CLEAR-REWRITE equals 2000 on each by issuing the **show module csm 2 vserver name ssl-rewrite detail** and **show module csm 2 vserver name clear-rewrite detail** commands.

Step 14 On the Active CSM verify traffic was evenly load balanced between all of the SSLMs in serverfarm SSLSM and serverfarm CLEAR-REWRITE by issuing the **show mod csm 2 real sfarm sslsm detail** and **show mod csm 2 real sfarm clear-rewrite detail** commands.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the SSLM can rewrite the server issued 300 series redirects from HTTP to HTTPS.

Results

URL Rewrite Service Switch passed with exception. The following exceptions were noted: CSCec74017 and CSCeh70549.

Redundancy

The resiliency of network resources and services to hardware and software component failures is important to a successful high availability strategy in a data center network. These tests measure the effects of various failure scenarios on Layer 4-7 services and hardware.

This section contains the following topics:

Bundle HSRP Failover Service Switch

This test verified HSRP failover when a system failure occurred. This test also verified that the HSRP **preempt** command worked when the system returned to an operational state, if the interface was configured with a higher priority than the current active router interface in the same HSRP group. HTTPS traffic was sent through an FWSM and load balanced via CSM and SSLM.

Test Procedure

The procedure used to perform the Bundle HSRP Failover Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the **show standby brief** command on both dcb-ss-1 and dcb-ss-2 to verify that dcb-ss-2 is active for all VLANs.

Step 3 On dcb-ss-1, issue the **show module csm 2 ft** command to verify that this CSM is active.

Step 4 On the FWSM contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130 in dcb-ss-1, issue the **show failover** system context command to verify that this FWSM is active for that context.

Step 5 Issue the **clear mod csm 2 count** command on the active CSM to clear the CSM counters. Issue the following commands to verify they have been cleared and to verify state:

```
show mod csm 2 vservers name vip1 detail
show mod csm 2 vservers name vip-passive-ftp detail
show mod csm 2 vservers name vip29 detail
show mod csm 2 vservers name ssl29 detail
show mod csm 2 vservers name vip30 detail
show mod csm 2 vservers name ssl30 detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-B detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 stats
```

Step 6 Issue the **show ssl-proxy service** command on all SSLSM's to verify the services are operational.

Step 7 Issue the **clear ssl-proxy stats service** command on all four SSLSMs to clear SSL-proxy service statistics, then issue the **show ssl-proxy stats service** command to verify they have been cleared. (Some counters might have incremented due to CSM probes.)

Step 8 Initiate 20 minutes' worth of HTTP, HTTPS and FTP client traffic.

Step 9 Issue the following commands to verify that traffic is flowing to all the VIPs and being load-balanced on the servers:

```
show mod csm 2 vservers name vip1 detail
show mod csm 2 vservers name vip-passive-ftp detail
show mod csm 2 vservers name vip29 detail
show mod csm 2 vservers name ssl29 detail
show mod csm 2 vservers name vip30 detail
show mod csm 2 vservers name ssl30 detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-B detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 stats
```

Step 10 Issue the following commands on all four SSLSM's to verify the conns attempted and conns completed counter has incremented and there are no errors:

```
show ssl-proxy stats service
show ssl-proxy stats ssl
```

Step 11 Issue the **reload** command on dcb-ss-1 to force a failover.

Step 12 Issue the **show standby brief** command on dcb-ss-2 to verify it is now the active HSRP router for all VLANs.

Step 13 Verify that the CSM in dcb-ss-2 is now active using the **show mod csm 2 ft** command.

Step 14 Verify that the FWSM in dcb-ss-2 is now active for contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130 using the **show failover** command in these contexts.

Step 15 Issue the **clear mod csm 2 count** command on the active CSM to clear the CSM counters. Issue the following commands to verify they have been cleared and to verify state:

```
show mod csm 2 vservers name vip1 detail
show mod csm 2 vservers name vip-passive-ftp detail
show mod csm 2 vservers name vip29 detail
show mod csm 2 vservers name ssl29 detail
show mod csm 2 vservers name vip30 detail
show mod csm 2 vservers name ssl30 detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-B detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 stats
```

Step 16 Issue the following commands on both remaining SSLSMs in dcb-ss-2 to verify the conns attempted and conns completed counter are still incrementing and there are no errors:

```
show ssl-proxy stats service
show ssl-proxy stats ssl
```

Step 17 When dcb-ss-1 becomes operational again issue the **show standby brief** command to verify it preempts and again becomes active HSRP router for all VLANs.

Step 18 Verify that the CSM in dcb-ss-1 has preempted and is again active using the **show mod csm 2 ft** command.

Step 19 Verify that the Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130 contexts for the FWSM in dcb-ss-2 is still active using the **show failover** command in these contexts.

Step 20 Issue the following commands on the CSM in dcb-ss-1 to verify that it is forwarding traffic to the VIPs and load-balancing traffic to the servers:

```
show mod csm 2 vservers name vip1 detail
show mod csm 2 vservers name vip-passive-ftp detail
show mod csm 2 vservers name vip29 detail
show mod csm 2 vservers name ssl29 detail
show mod csm 2 vservers name vip30 detail
show mod csm 2 vservers name ssl30 detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-B detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 stats
```

Step 21 Wait for the client traffic to stop, then report the results.

Step 22 Perform a manual failback of the active FWSM in dcb-ss-2, using the **fail active** command on the standby, so that the FWSM in dcb-ss-1 becomes active.

Step 23 Verify that the FWSM in dcb-ss-1 is now active for contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130 using the **show failover** command in these contexts.

Step 24 Stop background scripts to collect final status of network devices and analyze for error.

Step 25 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the services provided on the active HSRP router (CSM, FWSM, SSLM) will be failed over the the standby HSRP router properly.
- We expect the failed system to become operational again and resume HSRP active status and forward traffic.

Results

Bundle HSRP Failover Service Switch passed.

Bundle FWSM Redundancy Service Switch

This test verified that long-lived flows being load balanced by the CSM and traversing the FWSM will be replicated between the primary and secondary FWSM. The ability of the system to successfully replicate flows and forward traffic after the failover was the criteria for a successful test run.

Test Procedure

The procedure used to perform the Bundle FWSM Redundancy Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the following commands on the primary and secondary FWSM for contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130:

```
show xlate
show conn
```

Step 3 Issue the **show failover** command from the contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130 on the primary and secondary FSWM to verify the primary FWSM is in active state for those contexts.

Step 4 Issue the **clear mod csm 2 count** command on the active CSM to clear counters.

Step 5 Issue the following commands to verify the counters have been cleared:

```
show mod csm 2 vservers name ssl30 detail
show mod csm 2 vservers name ssl29 detail
```

```
show mod csm 2 vservers name vip-passive-ftp detail
show mod csm 2 vservers name vip1 detail
show mod csm 2 real sfarm sslsm detail
show mod csm 2 real sfarm farm1-b-be detail
show mod csm 2 real sfarm farm1-b-be detail
show mod csm 2 real sfarm farm1-b detail
show mod csm 2 stats
show mod csm 2 conn
```

Step 6 Generate HTTP traffic to vserver VIP1, HTTPS traffic to vservers SSL29 and SSL30, and FTP traffic to vserver VIP-PASSIVE-FTP.

Step 7 Issue the following commands on the primary and secondary FWSM for contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130:

```
show xlate
show conn
```

Step 8 Issue the following commands on the active CSM to verify connections:

```
show mod csm 2 vservers name ssl30 detail
show mod csm 2 vservers name ssl29 detail
show mod csm 2 vservers name vip-passive-ftp detail
show mod csm 2 vservers name vip1 detail
show mod csm 2 real sfarm sslsm detail
show mod csm 2 real sfarm farm1-b-be detail
show mod csm 2 real sfarm farm1-b-be detail
show mod csm 2 real sfarm farm1-b detail
show mod csm 2 stats
show mod csm 2 conn
```

Step 9 Issue the **reload** command on the primary FWSM to force a reload.

Step 10 Issue the **show failover** command on the secondary FSWM contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130 to verify it is now active for those contexts.

Step 11 Issue the following commands on the secondary FWSM to verify connections:

```
show xlate
show conn
```

Step 12 Issue the following commands on the active CSM several times to verify connections:

```
show mod csm 2 vservers name ssl30 detail
show mod csm 2 vservers name ssl29 detail
show mod csm 2 vservers name vip-passive-ftp detail
show mod csm 2 vservers name vip1 detail
show mod csm 2 real sfarm sslsm detail
show mod csm 2 real sfarm farm1-b-be detail
show mod csm 2 real sfarm farm1-b-be detail
show mod csm 2 real sfarm farm1-b detail
show mod csm 2 stats
show mod csm 2 conn
```

Step 13 When the failed FWSM comes back online issue the **show failover** command from the contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130 to verify it is in standby state for those contexts.

Step 14 Issue the following command on the now standby FWSM to verify connections have been replicated from the secondary FWSM:

```
show xlate
show conn
```

Step 15 Issue the **reload** command on the secondary FWSM to force a reload.

Step 16 Issue the **show failover** command on the primary FSWM from the contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130 to verify it is now active for those contexts.

Step 17 Issue the following commands on the primary and secondary FWSM for contexts Vlan1101-2101, Vlan1129-2129, and Vlan1130-2130:

```
show xlate
show conn
```

Step 18 Issue the following commands on the active CSM several times to verify connections:

```
show mod csm 2 vservers name ssl30 detail
show mod csm 2 vservers name ssl29 detail
show mod csm 2 vservers name vip-passive-ftp detail
show mod csm 2 vservers name vip1 detail
show mod csm 2 real sfarm sslsm detail
show mod csm 2 real sfarm farm1-b-be detail
show mod csm 2 real sfarm farm1-b-be detail
show mod csm 2 real sfarm farm1-b detail
show mod csm 2 stats
show mod csm 2 conn
```

Step 19 Wait for traffic to complete and record the results, checking for errors.

Step 20 Stop background scripts to collect final status of network devices and analyze for error.

Step 21 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the FWSM to replicate flow information from the active to standby FWSM.
- We expect the standby FWSM to transition to the active state with the failure of the active FWSM.

Results

Bundle FWSM Redundancy Service Switch passed with exception. The following exceptions were noted: CSCsj16292.

Bundle SSLSM Reset Service Switch

This test verified the effect of an SSL module reset on CSM load balancing. The CSM TCP probe detects the module failure and stops load balancing traffic to it. The CSM continued to load balancing traffic to the remaining operational SSL Module. When the CSM TCP probe detects the SSLM Module is operational again it will start to load balance traffic to it.

Test Procedure

The procedure used to perform the Bundle SSLSM Reset Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the **clear module csm 2 counters** command on dcb-ss-1 to clear the CSM counters.

Step 3 Issue the following commands to verify the counters have been cleared:

```
show mod csm 2 vservers name SSL30 detail
show mod csm 2 vservers name VIP30 detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 serverfarm name SSLSM detail
show mod csm 2 stats
```

Step 4 Issue the following commands on all four SSLMs to verify the services are operational:

```
show ssl-proxy service BACKENDCLIENT
show ssl-proxy service backend
show ssl-proxy service dcap-frontend
```

Step 5 Issue the following commands on all four SSLSMs to clear SSL-proxy service statistics:

```
clear ssl-proxy stats service BACKENDCLIENT
clear ssl-proxy stats service backend
clear ssl-proxy stats service dcap-backend
```

Step 6 Issue the following commands to verify they have been cleared:

```
show ssl-proxy stats service BACKENDCLIENT
show ssl-proxy stats service backend
show ssl-proxy stats service dcap-backend
```

Step 7 From several outside clients initiate a mixture of traffic including client requests involving both front-end encryption only (via CSM VIP SSL29) and front- and back-end encryption (via CSM VIP SSL30).

Step 8 When the traffic has stopped, issue the following commands on the active CSM to verify vservers SSL29, SSL30, VIP30, and VIP29 have opened connections:

```
show mod csm 2 vservers name SSL30 detail
show mod csm 2 vservers name VIP30 detail
show mod csm 2 vservers name SSL29 detail
show mod csm 2 vservers name VIP29 detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 serverfarm name SSLSM detail
show mod csm 2 stats
```

Step 9 Issue the following commands on all four SSLSMs to verify the conns attempted and conns completed counter has incremented and there are no errors:

```
show ssl-proxy stats service BACKENDCLIENT
show ssl-proxy stats service backend
show ssl-proxy stats service dcap-frontend
```

Step 10 Use the **no power enable module 3** command on dcb-ss-1 to remove power to dcb-ss-1-sslm-1.

Step 11 Verify that the health probe from the CSM to one of the real servers in serverfarm SSLSM fails after a time using the **show module csm 2 real sfarm sslsm det** command.

Step 12 Start another set of HTTPS client requests after clearing the counters on the CSM using the **clear module csm 2 counters** command.

Step 13 Issue the following commands on the active CSM to verify that all traffic is being load balanced to the other three SSLMs:

```
show mod csm 2 vservers name SSL30 detail
show mod csm 2 vservers name VIP30 detail
show mod csm 2 vservers name SSL29 detail
show mod csm 2 vservers name VIP29 detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 serverfarm name SSLSM detail
show mod csm 2 stats
```

Step 14 Issue the following commands on the remaining three SSLMs to verify the conns attempted and conns completed counter are incrementing and there are no errors:

```
show ssl-proxy stats service BACKENDCLIENT
show ssl-proxy stats service backend
show ssl-proxy stats service dcap-frontend
```

Step 15 After powering the SSLM back on and verifying its placement back in the serverfarm using the **show mod csm 2 real sfarm sslsm detail** command, start another set of HTTPS client requests.

Step 16 Issue the following commands to make sure traffic is again being load balanced among the four SSLMs:

```
show mod csm 2 vservers name SSL30 detail
show mod csm 2 vservers name VIP30 detail
show mod csm 2 vservers name SSL29 detail
show mod csm 2 vservers name VIP29 detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 serverfarm name SSLSM detail
show mod csm 2 stats
```

Step 17 Stop background scripts to collect final status of network devices and analyze for error.

Step 18 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the CSM TCP probe to detect the SSLM failure.
- We expect the CSM to reset open connections when a probe fails a real.
- We expect the CSM to properly load balance traffic around a failed SSLM.

Results

Bundle SSLSM Reset Service Switch passed with exception. The following exceptions were noted: CSCeh70549.

CSM Redundancy Service Switch

This test verified that flow information was replicated from the active CSM to the standby CSM. Upon a redundancy transition the standby CSM became the new active CSM and processed all flows that were originally created on the active CSM.

Test Procedure

The procedure used to perform the CSM Redundancy Service Switch test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Issue the **clear mod csm 2 counters** and **clear mod csm 2 sticky all** commands to clear CSM counters and sticky table on the active and standby CSM.

Step 3 Issue the following commands on the active and standby CSM to verify the counters have been cleared:

```
show mod csm 2 vservers name VIP29 detail
show mod csm 2 vservers name SSL29 detail
show mod csm 2 vservers name VIP30 detail
show mod csm 2 vservers name SSL30 detail
show mod csm 2 vservers name VIP1 detail
show mod csm 2 vservers name VIP-PASSIVE-FTP detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 4 Issue the **clear ssl-proxy stats service** command on all SSLMs to clear statistics.

Step 5 Issue the **show ssl-proxy service** command on all SSLMs to verify all proxy services are operational.

Step 6 Generate HTTPS and FTP traffic to vservers VIP1 and SSL29.

Step 7 Issue the following commands on the active CSM to verify traffic flow, and on the standby CSM to verify replication of connections and sticky information:

```
show mod csm 2 vservers name VIP29 detail
show mod csm 2 vservers name SSL29 detail
show mod csm 2 vservers name VIP30 detail
show mod csm 2 vservers name SSL30 detail
show mod csm 2 vservers name VIP1 detail
show mod csm 2 vservers name VIP-PASSIVE-FTP detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 8 Issue the **show ssl-proxy stats service** command on all SSLSM's to verify the conns completed counter has incremented and that there are no handshake failures.

Step 9 Issue the **hw-module module 2 reset** command to rest the active CSM in slot 2.

Step 10 Issue the **show mod csm 2 ft** command on the standby to verify it is now the active CSM.

Step 11 Issue the following commands on the newly active CSM to verify traffic flow:

```
show mod csm 2 vservers name VIP29 detail
show mod csm 2 vservers name SSL29 detail
show mod csm 2 vservers name VIP30 detail
show mod csm 2 vservers name SSL30 detail
show mod csm 2 vservers name VIP1 detail
show mod csm 2 vservers name VIP-PASSIVE-FTP detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 12 Issue the **show ssl-proxy stats service** command to verify the conns completed counter has incremented and that there are no handshake failures.

Step 13 When the reloaded CSM comes back online issue the **show mod csm 2 ft** command to verify it has preempted and is now the active CSM.

Step 14 Issue the following commands on the new active CSM to verify traffic flow:

```
show mod csm 2 vservers name VIP29 detail
show mod csm 2 vservers name SSL29 detail
show mod csm 2 vservers name VIP30 detail
show mod csm 2 vservers name SSL30 detail
show mod csm 2 vservers name VIP1 detail
show mod csm 2 vservers name VIP-PASSIVE-FTP detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 15 Issue the following commands on the standby CSM to verify traffic flow:

```
show mod csm 2 vservers name VIP29 detail
show mod csm 2 vservers name SSL29 detail
show mod csm 2 vservers name VIP30 detail
show mod csm 2 vservers name SSL30 detail
show mod csm 2 vservers name VIP1 detail
show mod csm 2 vservers name VIP-PASSIVE-FTP detail
show mod csm 2 real sfarm SSLSM detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B-BE detail
show mod csm 2 real sfarm FARM1-B detail
show mod csm 2 stats
show mod csm 2 sticky
show mod csm 2 conn
```

Step 16 Issue the **show ssl-proxy stats service** command to verify the conns completed counter has incremented and that there are no handshake failures.

Step 17 Wait for the traffic to complete and report the results.

Step 18 Stop background scripts to collect final status of network devices and analyze for error.

Step 19 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the CSM to replicate connections hitting the vserver.
- We expect the standby to become active and service the persistent replicated connection.
- We expect the CSM to preempt after a failure.
- We expect sticky connections to remain stuck to the same real after a failover.

Results

CSM Redundancy Service Switch passed.

CHAPTER 7

ACE

Tests in the "ACE" section on page 7-1focus on the functionality of the ACE service module, operating in the DCAP environment. Results for additional tests run by the Safe Harbor team against the software version tested in DCAP 4.0 are available in the DCAP 4.0 Appendix.

Refer to the following chapters for respective Layer 4-7 services testing:

- CSM—Based Integrated Switch Bundle "Layer 4-7 CSM" section on page 2-1
- ACE—Based Integrated Switch Bundle "Layer 4-7 ACE" section on page 3-1
- CSM—Based Service Chassis Bundle "Layer 4-7 Services Switch" section on page 4-1
- Intrusion Detection Services Module (IDSM) "IDSM IPS" section on page 6-1

Test Results Summary

Table 7-1 on page 7-2 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 7-1 on page 7-2 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 7-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Security, page 7-3	n/a	1. ACE IP Norm 2. ACE Oracle TCP Norm	
Service Load Balancing (SLB), page 7-10	n/a	1. ACE FTP 2. ACE Oracle Cookie Insert 3. ACE Oracle Header Insert 4. HTTP Inspection	CSCsh14278 CSCsl50509 CSCsg62851 CSCsl66036,
Traffic Handling, page 7-23	n/a	1. ACE Oracle RHI	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

- Security, page 7-3
- Service Load Balancing (SLB), page 7-10
- Traffic Handling, page 7-23

Security

The tests in this section look at some of the security features that the ACE module brings to the handling of data center traffic.

Security: TCP and IP Normalization

TCP normalization is a Layer 4 feature that consists of a series of checks that the ACE performs at various stages of a flow, from initial connection setup to the closing of a connection. The ACE uses these TCP connection settings to decide which checks to perform and whether to discard a TCP segment based on the results of the checks. The ACE discards segments that appear to be abnormal or malformed. IP normalization is used to protect itself and the data center from a variety of attacks. IP normalization is to Layer 3 what TCP normalization is to Layer 4. In general, IP normalization performs a series of checks on IP packets.

This section contains the following topics:

- ACE IP Norm, page 7-3
- ACE Oracle TCP Norm, page 7-6

ACE IP Norm

The ACE uses IP normalization to protect itself and the data center from a variety of attacks.

This test verified that the ACE takes appropriate actions based on the configured parameter settings.

```
parameter-map type connection NORM_IP_IND
    set ip tos 22
probe tcp TCP:8000
  port 8000
  interval 5
  faildetect 2
  passdetect interval 10
  open 3
serverfarm host ORAAPP_NORM_IND
  failaction purge
  predictor leastconns slowstart 10
  probe TCP:8000
  rserver ORAAPP01
    inservice
  rserver ORAAPP02
    inservice
  rserver ORAAPP03
    inservice
serverfarm host ORAAPP_L7_NORM2_IND
  failaction purge
```

```
  predictor leastconns slowstart 10
  probe TCP:8000
  retcode 100 599 check count
  rserver ORAAPP01 8000
    inservice
  rserver ORAAPP02 8000
    inservice
  rserver ORAAPP03 8000
    inservice
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  4 match request-method rfc get
  5 match request-method rfc head
  6 match request-method rfc options
  8 match request-method rfc put
  9 match request-method rfc trace
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
  30 match request-method rfc post
class-map match-any ORACLE_NORM_IND
  2 match virtual-address 101.1.33.65 tcp eq 8000
class-map match-any ORACLE_NORM2_L7_8888_8000_IND
  2 match virtual-address 101.1.33.65 tcp eq 8888
 class-map match-all NORM_ALL_TRAFFIC_IND
  2 match any
policy-map type loadbalance first-match ORAAPP_NORM_IND
  class class-default
  serverfarm ORAAPP_NORM_IND
policy-map type loadbalance first-match ORAAPP_L7_NORM2_IND
  class class-default
  serverfarm ORAAPP_L7_NORM2_IND
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
  class ORACLE_NORM_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_NORM_IND
    loadbalance vip icmp-reply active
  class ORACLE_NORM2_L7_8888_8000_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_L7_NORM2_IND
    loadbalance vip icmp-reply active
    inspect http policy INSPECT_GOOD_HTTP url-logging
policy-map multi-match NORMALIZATION_IND
   class NORM_ALL_TRAFFIC_IND
   connection advanced-options NORM_IP_IND
```

Test Procedure

The procedure used to perform the ACE IP Norm test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the primary CAT (dca-agg-1-ace-1), use the NAM to start a packet capture on the ACE's internal portchannel.

Step 3 From a Linux client, send a burst of SYNs with the IP type of service set to 'FF' to an L4 VIP and directly to a backend server. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 --tos ff -c 10 -V
hping2 101.1.33.16 -S -p 8000 --tos ff -c 10 -V
```

Step 4 Stop the packet capture and view the results. Verify that packets sent to the L4 VIP and directly to the server have an IP TOS value of 'FF' on both sides of the ACE, client and server vlans.

Step 5 Configure a global service policy to apply a parameter map across the whole context. This will cause the ACE to apply the parameters of this map for all traffic, routed or load-balanced. This parameter map will alter the IP TOS value in the packet to '0x16' when transmitted on the server vlan. Issue the following commands:

```
config
policy-map multi-match NORMALIZATION_IND
class NORM_ALL_TRAFFIC_IND
connection advanced-options NORM_IP_IND
exit
exit
service-policy input NORMALIZATION_IND
end
```

Step 6 On the primary CAT (dca-agg-1-ace-1), use the NAM to start a packet capture on the ACE's internal portchannel.

Step 7 From a Linux client, send a burst of SYNs with the IP type of service set to 'FF' to an L4 VIP and directly to a backend server. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 --tos ff -c 10 -V
hping2 101.1.33.16 -S -p 8000 --tos ff -c 10 -V
```

Step 8 Stop the packet capture and view the results. Verify that packets sent to the L4 VIP and directly to the server have an IP TOS value of 'FF' on the client side of the trace and '0x16' on the server side.

Step 9 On the primary CAT (dca-agg-1-ace-1), use the NAM to start a packet capture on the ACE's internal portchannel.

Step 10 From a Linux client, send a burst of SYNs with the IP TTL set to 15. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 --ttl 15 -c 10 -V
hping2 101.1.33.16 -S -p 8000 --ttl 15 -c 10 -V
```

Step 11 Stop the packet capture and view the results. Verify that packets sent to the L4 VIP and directly to the server have an IP TTL value less than 15 on both the client and server vlans.

Step 12 Configure the ACE to set a minimum TTL value of 40. Setting the minimum ttl to 40 will cause the ACE to change the ttl value to 40 if a packet is received with one that is less. Issue the following commands:

```
config
interface vlan 2132
```

```
ip ttl minimum 40
end
```

Step 13 On the primary CAT (dca-agg-1-ace-1), use the NAM to start a packet capture on the ACE's internal portchannel.

Step 14 From a Linux client, send a burst of SYNs with the IP TTL set to 15. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 --ttl 15 -c 10 -V
hping2 101.1.33.16 -S -p 8000 --ttl 15 -c 10 -V
```

Step 15 Stop the packet capture and view the results. Verify that the packets sent have a TTL value less than 15 on the client side and 40 on the server side.

Step 16 Return the config back to its original configuration. Issue the following commands:

```
config
no service-policy input NORMALIZATION_IND
interface vlan 2132
no ip ttl minimum 40
end
```

Step 17 Stop background scripts to collect final status of network devices and analyze for error.

Step 18 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the ACE to apply a service policy globally that applies to all traffic passing through the ACE.
- We expect that the ACE will alter IP values in the packets as configured.
- We expect no CPU or memory problems.

Results

ACE IP Norm passed.

ACE Oracle TCP Norm

TCP normalization is a Layer 4 feature that consists of a series of checks that the ACE performs at various stages of a flow, from initial connection setup to the closing of a connection.You can control many of the segment checks by configuring one or more advanced TCP connection settings. The ACE uses these TCP connection settings to decide which checks to perform and whether to discard a TCP segment based on the results of the checks. The ACE discards segments that appear to be abnormal or malformed.

This test verified that the ACE takes appropriate actions based on the configured parameter settings.

```
parameter-map type connection NORM_TCP_IND
    tcp-options timestamp allow
    syn-data drop
    reserved-bits clear
probe tcp TCP:8000
  port 8000
```

```
  interval 5
  faildetect 2
  passdetect interval 10
  open 3
serverfarm host ORAAPP_NORM_IND
  failaction purge
  predictor leastconns slowstart 10
  probe TCP:8000
  rserver ORAAPP01
    inservice
  rserver ORAAPP02
    inservice
  rserver ORAAPP03
    inservice
serverfarm host ORAAPP_L7_NORM2_IND
  failaction purge
  predictor leastconns slowstart 10
  probe TCP:8000
  retcode 100 599 check count
  rserver ORAAPP01 8000
    inservice
  rserver ORAAPP02 8000
    inservice
  rserver ORAAPP03 8000
    inservice
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  4 match request-method rfc get
  5 match request-method rfc head
  6 match request-method rfc options
  8 match request-method rfc put
  9 match request-method rfc trace
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
  30 match request-method rfc post
class-map match-any ORACLE_NORM_IND
  2 match virtual-address 101.1.33.65 tcp eq 8000
class-map match-any ORACLE_NORM2_L7_8888_8000_IND
  2 match virtual-address 101.1.33.65 tcp eq 8888
 class-map match-all NORM_ALL_TRAFFIC_IND
  2 match any
policy-map type loadbalance first-match ORAAPP_NORM_IND
  class class-default
  serverfarm ORAAPP_NORM_IND
policy-map type loadbalance first-match ORAAPP_L7_NORM2_IND
  class class-default
```

```
  serverfarm ORAAPP_L7_NORM2_IND
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
  class ORACLE_NORM_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_NORM_IND
    loadbalance vip icmp-reply active
  class ORACLE_NORM2_L7_8888_8000_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_L7_NORM2_IND
    loadbalance vip icmp-reply active
    inspect http policy INSPECT_GOOD_HTTP url-logging
policy-map multi-match NORMALIZATION_IND
   class NORM_ALL_TRAFFIC_IND
   connection advanced-options NORM_TCP_IND
```

Test Procedure

The procedure used to perform the ACE Oracle TCP Norm test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Clear the normalization me-stats counters by running the following show commands:

```
show np 1 me-stats "-M 1 -s norm"
show np 2 me-stats "-M 1 -s norm"
```

Step 3 From a Linux client, send a burst of SYNs with data to an L4, L7 VIP and directly to a backend server. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 -d 512 -c 10 -V
hping2 101.1.33.65 -S -p 8888 -d 512 -c 10 -V
hping2 101.1.33.16 -S -p 8000 -d 512 -c 10 -V
```

Step 4 By default the ACE allows a SYN packet with data, responses packets should be seen on the client's console and the normalization statistics should show no 'syn data denied' counted. Issue the following commands:

```
show np 1 me-stats "-M 1 -s norm" | include syn
show np 2 me-stats "-M 1 -s norm" | include syn
```

Step 5 From a Linux client, send a burst of SYNs with the timestamp option to an L4, L7 VIP and directly to a backend server. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 --tcp-timestamp -c 10 -V
hping2 101.1.33.65 -S -p 8888 --tcp-timestamp -c 10 -V
hping2 101.1.33.16 -S -p 8000 --tcp-timestamp -c 10 -V
```

Step 6 By default the ACE clears the timestamp option from the SYN, so no timestamp responses should be seen on the client's console.

```
show np 1 me-stats "-M 1 -s norm" | include timestamp
show np 2 me-stats "-M 1 -s norm" | include timestamp
```

Step 7 From a Linux client, send a burst of SYNs with the reserved bits set to an L4, L7 VIP and directly to a backend server. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 -X -Y -c 10 -V
hping2 101.1.33.65 -S -p 8888 -X -Y -c 10 -V
hping2 101.1.33.16 -S -p 8000 -X -Y -c 10 -V
```

Step 8 By default the ACE allows the reserved bits, so responses should be seen on the client's console and the normalization statistics should be zero for 'reserved' counters. Issue the following commands:

```
show np 1 me-stats "-M 1 -s norm" | include reserved
show np 2 me-stats "-M 1 -s norm" | include reserved
```

Step 9 Configure a global service policy to apply a parameter map across the whole context. This will cause the ACE to apply the parameters of this map for all traffic, routed or load-balanced. Issue the following commands:

```
config
policy-map multi-match NORMALIZATION_IND
class NORM_ALL_TRAFFIC_IND
connection advanced-options NORM_TCP_IND
exit
exit
service-policy input NORMALIZATION_IND
end
```

Step 10 From a Linux client, send a burst of SYNs with data to an L4, L7 VIP and directly to a backend server. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 -d 512 -c 10 -V
hping2 101.1.33.65 -S -p 8888 -d 512 -c 10 -V
hping2 101.1.33.16 -S -p 8000 -d 512 -c 10 -V
```

Step 11 The ACE is configured to drop SYN packets that contain data, so no responses packets should be seen on the client's console and the normalization statistics should show 'syn data denied' counted. Issue the following commands:

```
show np 1 me-stats "-M 1 -s norm" | include syn
show np 2 me-stats "-M 1 -s norm" | include syn
```

Step 12 From a Linux client, send a burst of SYNs with the timestamp option to an L4, L7 VIP and directly to a backend server. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 --tcp-timestamp -c 10 -V
hping2 101.1.33.65 -S -p 8888 --tcp-timestamp -c 10 -V
hping2 101.1.33.16 -S -p 8000 --tcp-timestamp -c 10 -V
```

Step 13 The ACE is configured to allow the timestamp option, so timestamp responses should be seen on the client's console.

Step 14 From a Linux client, send a burst of SYNs with the reserved bits set to an L4, L7 VIP and directly to a backend server. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 -X -Y -c 10 -V
hping2 101.1.33.65 -S -p 8888 -X -Y -c 10 -V
hping2 101.1.33.16 -S -p 8000 -X -Y -c 10 -V
```

Step 15 The ACE is configured to clear the reserved bits, so the normalization statistics should show 'TCP zeroed reserved field' counted. Issue the following commands:

```
show np 1 me-stats "-M 1 -s norm" | include reserved
show np 2 me-stats "-M 1 -s norm" | include reserved
```

Step 16 Change the parameter map to drop packets with the reserved bits set. Issue the following commands:

```
config
parameter-map type connection NORM_TCP
reserved-bits drop
```

Step 17 From a Linux client, send a burst of SYNs with the reserved bits set to an L4, L7 VIP and directly to a backend server. Issue the following commands one at a time:

```
hping2 101.1.33.65 -S -p 8000 -X -Y -c 10 -V
hping2 101.1.33.65 -S -p 8888 -X -Y -c 10 -V
hping2 101.1.33.16 -S -p 8000 -X -Y -c 10 -V
```

Step 18 The ACE is configured to drop packets with the reserved bits set, so no responses should be seen on the client's console and the normalization statistics should show '[Drops] TCP non-zero reserved field' counted. Issue the following commands:

```
show np 1 me-stats "-M 1 -s norm" | include reserved
show np 2 me-stats "-M 1 -s norm" | include reserved
```

Step 19 Return the config back to its original configuration. Issue the following commands:

```
config
parameter-map type connection NORM_TCP
reserved-bits clear
exit
no service-policy input NORMALIZATION_IND
end
```

Step 20 Stop background scripts to collect final status of network devices and analyze for error.

Step 21 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the ACE to apply a service policy globally that applies to all traffic passing through the ACE.
- We expect that the ACE will drop, clear or allow packets in accordance with the normalization configuraton.
- We expect no CPU or memory problems.

Results

ACE Oracle TCP Norm passed.

Service Load Balancing (SLB)

The tests in this section focus on some of the basic and advanced server load balancing functions of the ACE module.

File Transfer Protocol (FTP) inspection inspects FTP sessions for address translation in a message, dynamic opening of ports, stateful tracking of request and response messages. Each specified FTP command must be acknowledged before the ACE allows a new command. Command filtering allows you to restrict specific commands by the ACE. When the ACE denies a command, it closes the connection. The ACE performs a stateful deep packet inspection of the HTTP protocol. Deep packet inspection is a special case of application inspection where the ACE examines the application payload of a packet or a traffic stream and makes decisions based on the content of the data. During HTTP deep inspection, the main focus of the application inspection process is on HTTP attributes such as HTTP header, URL, and to a limited extent, the payload. The ACE performs regular expression matching against the received packet data from a particular connection based on the cookie expression.

This section contains the following topics:

ACE FTP

The FTP Inspection feature on ACE inspects FTP sessions for address translation in a message, dynamic opening of ports, stateful tracking of request and response messages. Each specified FTP command must be acknowledged before the ACE allows a new command. Command filtering allows you to restrict specific commands by the ACE. When the ACE denies a command, it closes the connection using a TCP Reset. This test will ensure that the FTP Inspection feature functioned as expected when denying specific FTP client commands.

Test Procedure

The procedure used to perform the ACE FTP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Clear the appropriate ftp inspection counters on the ACE by issuing the **clear stats inspect** commands:

Step 3 On the primary CAT (dca-agg-1-ace-1), use the NAM to start a packet capture on the ACE's internal portchannel.

Step 4 FTP to 101.1.33.210 (root/rebornagain) and GET the file 2M-1.doc

Step 5 Verify on the NAM that you see the ftp session and that you are using active/PORT mode ftp and that the file downloads completely.

Step 6 On the ACE, configure the following policy map: **policy-map type inspect ftp first-match FTP_INSPECT class FTP_INSPECT deny**

Step 7 On the ACE, issue the **show stats inspect inspect ftp** command in order to view the ftp inspection statitics. **dca-agg-1-ace-1/c2# show stats inspect ftp**

Step 8 On the primary CAT (dca-agg-1-ace-1), use the NAM to start a packet capture on the ACE's internal portchannel.

Step 9 Verify on the NAM that you see a TCP RESET being sent to the client from the VIP on ACE in response to the HTTP FTP request being sent from the client.

Step 10 Clear the appropriate ftp inspection counters on the ACE by issuing the **clear stats inspect** commands:

Step 11 Start another packet capture on the ACE's internal portchannel via the NAM.

Step 12 Open a browser and FTP to 101.1.33.210 (root/rebornagain) and get(copy) the file file 2M-1.doc

Step 13 Verify on the NAM that you now see the ftp session and that you are now using passive mode ftp (rather than active mode which was used previously)and that the file downloads completely.

Step 14 policy-map type inspect ftp first-match FTP_INSPECT class FTP_INSPECT deny

Step 15 Verify on the NAM that you see a TCP RESET being sent to the client from the VIP on ACE in response to the FTP GET request being sent from the client.

Step 16 Start another packet capture on the ACE's internal portchannel via the NAM.

Step 17 Open a browser and FTP to 101.1.33.12 (root/rebornagain)

Step 18 View the trace on the NAM in order to see the response to the ftp SYST command sent from the client.

Step 19 Start another packet capture on the ACE's internal portchannel via the NAM.

Step 20 Open a browser and FTP to 101.1.33.210 (root/rebornagain)

Step 21 View the trace on the NAM in order to see the response to the ftp SYST command sent from the client.

Step 22 Stop background scripts to collect final status of network devices and analyze for error.

Step 23 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the ACE to deny FTP client GET requests based on the configuration of the FTP Inspection feature.
- We expect the ACE to send the client a TCP RESET in response to an FTP GET when the FTP Inspection policy deny has been matched.
- We expect no CPU or memory problems.

Results

ACE FTP passed.

ACE Oracle Cookie Insert

Cookie sticky allows HTTP connections to remain stuck to a specific server until the cookie or sticky timer expires. The cookie insert feature is used when you want to use a session cookie for persistence if the server is not currently setting the appropriate cookie. With this feature enabled, the ACE inserts the cookie in the server response to the client. This allows the client to remain stuck to the same server.

This test verified that HTTP connections remain stuck to a particular server from an ACE injected cookie contained in the HTTP header.

Relevant ACE Configuration:

```
parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
rserver host ORAAPP01
  description ORACLE_APP01
  ip address 101.1.33.16
  inservice
rserver host ORAAPP02
  description ORACLE_APP02
  ip address 101.1.33.5
  inservice
rserver host ORAAPP03
  description ORACLE_APP03
  ip address 101.1.33.47
  inservice
rserver redirect REDIRECT_80_TO_8000
```

```
  description 302 REDIRECT TO PORT 80
  webhost-redirection http://wwwin-oefin.gslb.dcap.com:8000 302
  inservice
serverfarm host ORAAPP_CINSERT_IND
  description ORACLE WEB AND DB SERVERS
  failaction purge
  probe ORACLE_WEB_PAGE_CHECK_IND
  rserver ORAAPP01
    inservice
  rserver ORAAPP02
    inservice
  rserver ORAAPP03
    inservice
serverfarm redirect REDIRECT_PORT_80_TO_PORT_8000
  description REDIRECT CLIENTS FROM 80 To 8000
  failaction purge
  rserver REDIRECT_80_TO_8000
    inservice
sticky http-cookie ACE_COOKIE ORAAPP_CINSERT_IND
  cookie insert browser-expire
  replicate sticky
  serverfarm ORAAPP_CINSERT_IND
sticky http-cookie ACE_COOKIE ORAAPP_CINSERT2_IND
  cookie insert browser-expire
  replicate sticky
  serverfarm ORAAPP_CINSERT_IND
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  4 match request-method rfc get
  5 match request-method rfc head
  6 match request-method rfc options
  8 match request-method rfc put
  9 match request-method rfc trace
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
  30 match request-method rfc post
class-map type http loadbalance match-any L7_ORACLE
  description REGEX URL'S FOR ORACLE
  2 match http url /apptitle.html
  3 match http url /applist.html
  4 match http url /appdet.html
  5 match http url /appsmed3.gif
  6 match http url /aplogon.html
  7 match http url /oa_servlets/AppsLogin/.*
  8 match http url /OA_HTML/.*
  9 match http url /OA_HTML/fndvald.jsp
```

```
  10 match http url /pls/OEFIN/.*
  11 match http url /dev60cgi/.*
  12 match http url /OA_JAVA/oracle/apps/fnd/jar/.*
  13 match http url /OA_JAVA/java/awt/.*
  14 match http url /OA_MEDIA/appslogo_new.gif
  15 match http url /OA_JAVA/oracle/.*
  16 match http url /forms/.*
  17 match http url /OA_JAVA/oracle/apps/fnd/jar/fndutil.jar
  18 match http url /
  19 match http url /oa_servlets/.*
  20 match http url /OA_MEDIA/.*
  21 match http url /.*
  22 match http url .*
class-map type http loadbalance match-all ONLY_VALID_HTTP_FOR_REDIRCT_CANDIDATE
  2 match http header Host header-value "wwwin-oefin.gslb.dcap.com"
  3 match http url / method GET
class-map match-any ORACLE_L4_NO_WAAS_CINSERT2_IND
  2 match virtual-address 101.1.33.63 tcp eq 8000
class-map match-any ORACLE_L4_NO_WAAS_CINSERT_IND
  2 match virtual-address 101.1.33.62 tcp eq 8000
class-map match-all REDIRECT_VIP_L4_NO_WAAS_CINSERT2_IND
  2 match virtual-address 101.1.33.63 tcp eq www
class-map match-all REDIRECT_VIP_L4_NO_WAAS_CINSERT_IND
  2 match virtual-address 101.1.33.62 tcp eq www
policy-map type loadbalance first-match ORAAPP_SERVERS_CINSERT2_IND
  class class-default
    sticky-serverfarm ORAAPP_CINSERT2_IND
policy-map type loadbalance first-match ORAAPP_SERVERS_CINSERT_IND
  class L7_ORACLE
    sticky-serverfarm ORAAPP_CINSERT_IND
policy-map type loadbalance first-match REDIRECT_TCP:80_TO_TCP:8000_2_IND
  class class-default
    serverfarm REDIRECT_PORT_80_TO_PORT_8000_IND
policy-map type loadbalance first-match REDIRECT_TCP:80_TO_TCP:8000_IND
  class ONLY_VALID_HTTP_FOR_REDIRCT_CANDIDATE
    serverfarm REDIRECT_PORT_80_TO_PORT_8000_IND
policy-map type inspect http all-match INSPECT_GOOD_HTTP
  class INSPECT_HTTP_GOOD
    permit
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
  class ORACLE_L4_NO_WAAS_CINSERT_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_SERVERS_CINSERT_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    inspect http policy INSPECT_GOOD_HTTP url-logging
  class REDIRECT_VIP_L4_NO_WAAS_CINSERT_IND
    loadbalance vip inservice
    loadbalance policy REDIRECT_TCP:80_TO_TCP:8000_IND
    loadbalance vip icmp-reply active
    inspect http policy INSPECT_GOOD_HTTP url-logging
  class ORACLE_L4_NO_WAAS_CINSERT2_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_SERVERS_CINSERT2_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
  class REDIRECT_VIP_L4_NO_WAAS_CINSERT2_IND
    loadbalance vip inservice
    loadbalance policy REDIRECT_TCP:80_TO_TCP:8000_2_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
```

Test Procedure

The procedure used to perform the ACE Oracle Cookie Insert test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On a Windows client verify that the MTU is set to 150 bytes to simulate a dial-up or satellite user. This will cause the HTTP headers to span multiple packets.

Step 3 Clear the counters on the ACE by issuing the following commands:

```
clear serverfarm ORAAPP_CINSERT_IND
clear stats http
```

Step 4 Send an HTTP GET request from a IE browser to http://wwwin-oefin.gslb.dcap.com/.

Step 5 Verify on the ACE that only one rserver received the requests and that a corresponding number of headers were inserted compared to the total connections. Issue the following commands:

```
show serverfarm ORAAPP_CINSERT_IND detail
show stats http | inc insert
```

Step 6 Using the browser session that was opened click on the following links, pausing briefly (this will allow TCP connections to timeout, ~30 seconds) for each one.

```
Apps Logon Links
E-Business Home Page      (login sysadmin/sysadmin)
System Administrator      (Near bottom left, *NOT System Administration)
Requests                  (Near top center, under Concurrent)
```

Step 7 When the pages have fully loaded there will be a form to perform a query. Click on 'Specific Requests', and then enter some text into the fields and click find. This will send an HTTP POST request to the server.

Step 8 When the query ends (no records will be returned) close out the Oracle application and then the browser.

Step 9 Verify on the ACE that only one rserver received all of these requests and that a corresponding number of headers were inserted. Issue the following commands:

```
show serverfarm ORAAPP_CINSERT_IND detail
show stats http | inc insert
```

Step 10 Repeat the prior steps this time using Firefox as the browser. Verify that the behavior remains the same.

Step 11 Clear the counters on the ACE by issuing the following commands:

```
clear serverfarm ORAAPP_CINSERT_IND
clear stats http
```

Step 12 From a Windows client start a packet capture listening on port 8000. Send an HTTP GET request from an IE browser to http://wwwin-oefin.gslb.dcap.com/.

Step 13 Using the browser session that is open click on the following links:

```
Apps Logon Links
E-Business Home Page      (login sysadmin/sysadmin)
System Administrator      (Near bottom left, *NOT System Administration)
```

Step 14 The ACE injected cookie is configured to expire when the browser session ends and only inject on the first server response. Stop the packet capture. Verify that the ACE inserts a cookie named 'ACE_COOKIE' on the first server response without an expiration date set. Verify that on a single TCP connection that subsequent server responses do not contain an ACE injected cookie.

Step 15 Clear the counters on the ACE by issuing the following commands:

```
clear serverfarm ORAAPP_CINSERT_IND
clear stats http
```

Step 16 Modify the sticky timeout to 60 minutes and add a parameter map using persistence rebalance. This will cause the ACE to set an expiration time and to inject a cookie on all server responses. Issue the following commands:

```
config
sticky http-cookie ACE_COOKIE ORAAPP_CINSERT_IND
cookie insert
timeout 60
exit
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
class ORACLE_L4_NO_WAAS_CINSERT_IND
appl-parameter http advanced-options HTTP_PARAM
end
```

Step 17 From a Windows client start a packet capture listening on port 8000. Send an HTTP GET request from a IE browser to http://wwwin-oefin.gslb.dcap.com/.

Step 18 Using the browser session that is open click on the following links:

```
Apps Logon Links
E-Business Home Page      (login sysadmin/sysadmin)
System Administrator      (Near bottom left, *NOT System Administration)
```

Step 19 The ACE injected cookie is configured to expire in one hour EST and to inject on all server responses. Stop the packet capture. Verify that the ACE inserts a cookie named 'ACE_COOKIE' on all server responses with the proper expiration date set.

Step 20 Verify on the ACE that only one rserver received the requests and that a greater number of headers were inserted than total connections. Issue the following commands:

```
show serverfarm ORAAPP_CINSERT_IND detail
show stats http | inc insert
```

Step 21 From a Linux client use CURL to send a series (6 times) of GETs to VIP 101.1.33.62. Verify from the tool's terminal output that six requests produce three unique cookie values twice.

Step 22 Remove the cookie insert feature from a different sticky group that is using a shared serverfarm. Issue the following commands:

```
config
sticky http-cookie ACE_COOKIE ORAAPP_CINSERT2_IND
no cookie insert browser-expire
end
```

Step 23 From a Linux client use CURL to send a series (6 times) of GETs to VIP 101.1.33.62. Verify from the tool's terminal output that six requests produce three unique cookie values twice.

Step 24 Return the config back to the default by removing the sticky timeout and parameter map that was added. Issue the following commands:

```
config
sticky http-cookie ACE_COOKIE ORAAPP_CINSERT_IND
no timeout 60
cookie insert browser-expire
exit
sticky http-cookie ACE_COOKIE ORAAPP_CINSERT2_IND
cookie insert browser-expire
exit
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
```

```
class ORACLE_L4_NO_WAAS_CINSERT_IND
no appl-parameter http advanced-options HTTP_PARAM
end
```

Step 25 Stop background scripts to collect final status of network devices and analyze for error.

Step 26 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the client traffic will be stuck to a single server through a series of GETs and POSTs during a browser session.
- We expect that the ACE will inject cookies in all server responses when configured properly.
- We expect that the ACE will inject cookies with the proper expiration date set as configured
- We expect no CPU or memory problems.

Results

ACE Oracle Cookie Insert failed. The following failures were noted: CSCsh14278, CSCsl50509, CSCsg62851.

ACE Oracle Header Insert

The HTTP header insert feature provides the ACE with the ability to insert information, such as the client IP, destination IP, custom, and other types into the HTTP header. This feature is useful in situations where the ACE is performing source NAT and the server application requires visibility to the original source IP. The ACE performs the header insert function in the client-to-server direction. This test verified that ACE inserted the configured headers when GETs and POSTs were issued, even when these requests spanned many packets.

Relevant ACE Configuration:

```
parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
probe tcp TCP:8000
  port 8000
  interval 5
  faildetect 2
  passdetect interval 10
  open 3
serverfarm host ORAAPP_HDR_INS_IND
  description ORACLE WEB AND DB SERVERS
  failaction purge
  probe TCP:8000
  retcode 100 599 check count
  rserver ORAAPP01
    inservice
  rserver ORAAPP02
    inservice
  rserver ORAAPP03
    inservice
serverfarm host ORAAPP_HDR_INS2_IND
```

```
  description ORACLE WEB AND DB SERVERS
  failaction purge
  probe TCP:8000
  retcode 100 599 check count
  rserver ORAAPP01 8000
    inservice
  rserver ORAAPP02 8000
    inservice
  rserver ORAAPP03 8000
    inservice
sticky ip-netmask 255.255.255.255 address source HDR_INS_SRCIP_STKY_IND
  replicate sticky
  timeout 30
  serverfarm ORAAPP_HDR_INS_IND
sticky ip-netmask 255.255.255.255 address source HDR_INS_SRCIP_STKY2_IND
  replicate sticky
  timeout 30
  serverfarm ORAAPP_HDR_INS2_IND
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  4 match request-method rfc get
  5 match request-method rfc head
  6 match request-method rfc options
  8 match request-method rfc put
  9 match request-method rfc trace
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
  30 match request-method rfc post
class-map type http loadbalance match-any L7_ORACLE
  description REGEX URL'S FOR ORACLE
  2 match http url /apptitle.html
  3 match http url /applist.html
  4 match http url /appdet.html
  5 match http url /appsmed3.gif
  6 match http url /aplogon.html
  7 match http url /oa_servlets/AppsLogin/.*
  8 match http url /OA_HTML/.*
  9 match http url /OA_HTML/fndvald.jsp
  10 match http url /pls/OEFIN/.*
  11 match http url /dev60cgi/.*
  12 match http url /OA_JAVA/oracle/apps/fnd/jar/.*
  13 match http url /OA_JAVA/java/awt/.*
  14 match http url /OA_MEDIA/appslogo_new.gif
  15 match http url /OA_JAVA/oracle/.*
  16 match http url /forms/.*
  17 match http url /OA_JAVA/oracle/apps/fnd/jar/fndutil.jar
```

```
  18 match http url /
  19 match http url /oa_servlets/.*
  20 match http url /OA_MEDIA/.*
  21 match http url /.*
  22 match http url .*
class-map match-any ORACLE_HDR_INS_IND
  2 match virtual-address 101.1.33.64 tcp eq 8000
class-map match-any ORACLE_HDR_INS2:8888_8000_IND
  2 match virtual-address 101.1.33.64 tcp eq 8888
policy-map type loadbalance first-match ORAAPP_HDR_INS_IND
  class L7_ORACLE
    sticky-serverfarm HDR_INS_SRCIP_STKY_IND
    insert-http
Custom-header_maxsize-255bytes_abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQRST
UVWXYZ_abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQRSTUVWXYZ_abcdefghijklmnopq
rstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQRSTUVWXYZ_abcdefghijklmnopq_255bytes header-value
"Maximum-size of inserted header value is 255 bytes
abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQRSTUVWXYZ_abcdefghijklmnopqrstuvwx
yz-01234567890_ABCDEFGHIJKLMNOP_This header’s value is 255bytes including the quotes used
in the config_255bytes"
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
policy-map type loadbalance first-match ORAAPP_HDR_INS2_IND
  class L7_ORACLE
    sticky-serverfarm HDR_INS_SRCIP_STKY2_IND
    insert-http
Custom-header_maxsize-255bytes_abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQRST
UVWXYZ_abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQRSTUVWXYZ_abcdefghijklmnopq
rstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQRSTUVWXYZ_abcdefghijklmnopq_255bytes header-value
"Maximum-size of inserted header value is 255 bytes
abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQRSTUVWXYZ_abcdefghijklmnopqrstuvwx
yz-01234567890_ABCDEFGHIJKLMNOP_This header’s value is 255bytes including the quotes used
in the config_255bytes"
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
  class ORACLE_HDR_INS_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_HDR_INS_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    inspect http policy INSPECT_GOOD_HTTP url-logging
  class ORACLE_HDR_INS2:8888_8000_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_HDR_INS2_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 1133
    inspect http policy INSPECT_GOOD_HTTP url-logging
    appl-parameter http advanced-options HTTP_PARAM
interface vlan 1133
  description SERVER_VLAN
  bridge-group 10
  ip options allow
  mtu 2000
  no normalization
  fragment min-mtu 68
  no icmp-guard
  access-group input BPDU-ALLOW
  access-group input anyone
```

```
access-group output anyone
nat-pool 1 101.1.33.70 101.1.33.70 netmask 255.255.255.0 pat
service-policy input NAT_POLICY
service-policy input REMOTE-MGNT
service-policy input ORACLE_TCP_TRAFFIC
service-policy input ORACLE_TCP_TRAFFIC_IND
service-policy input ORACLE_RHI_90-150_IND
no shutdown
```

Test Procedure

The procedure used to perform the ACE Oracle Header Insert test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On a Windows client verify that the MTU is set to 150 bytes to simulate a dial-up or satellite user. This will cause the HTTP headers to span multiple packets.

Step 3 Clear the HTTP counters on the ACE by issuing the **clear stats http** commands:

Step 4 On the primary CAT (dca-agg-1-ace-1), use the NAM to start a packet capture on the ACE's internal portchannel.

Step 5 Send an HTTP GET request from an IE and Firefox browser to http://wwwin-oefin.gslb.dcap.com:8000/.

Step 6 With each open browser session click on the following links:

```
Apps Logon Links
E-Business Home Page      (login sysadmin/sysadmin)
System Administrator      (Near bottom left, *NOT System Administration)
```

Step 7 Stop the packet capture, close the browser windows, and view the results.

By default the ACE will insert HTTP headers only on the first request of a persistent connection, on the server side of the trace. Isolate a GET request on two persistent TCP streams, one from each browser. Verify on both that only the first request has the configured headers inserted and are accurate, such as header names and values.

Step 8 Isolate a POST request on two persistent TCP streams, one from each browser. Verify on both that only the first request has the configured headers inserted and are accurate, such as header names and values. The output will only show a single POST followed by GETs for each browser stream, this is expected.

Step 9 Verify that there were no header insert errors seen by issuing the **show stats http | inc error** command.

Step 10 Clear the HTTP counters on the ACE by issuing the **clear stats http** commands:

Step 11 Start a packet capture on the ACE's internal portchannel.

Step 12 This time use the same VIP address, but listening on a different destination port, set to inject on all requests, and NAT the source IP. The ACE will destination port NAT this from 8888 to 8000 and will alter the original source IP, while injecting all of the headers on all server side GETs or POSTs. Send an HTTP GET request from an IE and Firefox browser to http://wwwin-oefin.gslb.dcap.com:8888/.

Step 13 With each open browser session click on the following links:

```
Apps Logon Links
```

Step 14 Stop the packet capture, close the browser windows, and view the results.

This time the ACE will insert HTTP headers on all requests of a persistent connection, on the server side of the trace. Isolate a GET request on two persistent TCP streams, one from each browser. Verify on both that all requests have the configured headers inserted and are accurate, such as header names and values.

Step 15 Verify that there were no header insert errors seen by issuing the **show stats http | inc error** command.

Step 16 Clear the HTTP counters on the ACE by issuing the **clear stats http** commands:

Step 17 Launch a continuous stream of traffic to the vserver 101.1.33.64:8888.

Step 18 Let the test traffic run for at least 10 minutes while periodically executing the following commands:

```
show clock
show stats http | include inserted
show stats http | include errors
```

Step 19 Verify that there were no header insert errors seen during this time period and then kill the client generated traffic.

Step 20 Start a packet capture on the ACE's internal portchannel.

Step 21 On a Linux client use CURL to send a couple of HTTP streams, one issuing GETs and the other POSTs on persistent connections.

Step 22 Stop the packet capture and view the results.

Isolate the persistent GET and POST requests. Verify on both that all requests have the configured headers inserted and are accurate, such as header names and values.

Step 23 Stop background scripts to collect final status of network devices and analyze for error.

Step 24 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the ACE to insert the configured headers on HTTP requests that span a number packets.
- We expect the ACE to insert the configured headers on all GET and POST request's within a persistent connection when properly configured.
- We expect no CPU or memory problems.

Results

ACE Oracle Header Insert passed.

HTTP Inspection

Deep packet inspection allows the ACE to examine the application payload of a packet or a traffic stream and make decisions based on the content of the data. During HTTP deep packet inspection, the main focus of the application inspection process is on HTTP attributes such as HTTP header, URL, and to a limited extent, the payload. This test will examine the http inspection feature on ACE by parsing both content length as well as http method's in order to permit or deny http traffic. An MTU of 576 is used at some points in the test in order to simulate a slow or dial-up user's traffic. WAAS is enabled for all HTTP traffic in the Data Center.

Test Procedure

The procedure used to perform the HTTP Inspection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Clear the appropriate http inspection counters on the ACE by issuing the **clear stats inspect** commands:

Step 3 On the primary CAT (dca-agg-1-ace-1), use the NAM to start a packet capture on the ACE's internal portchannel.

Step 4 From a branch client, Send an HTTP request to http://wwwin-oefin.gslb.dcap.com:8000/OA_HTML/AppsLocalLogin.jsp?requestUrl=APPSHOMEPAGE&cancelUrl=http%3A%2F%2Fwwwin-oefin.gslb.dcap.com%3A8000%2Foa_servlets%2Foracle.apps.fnd.sso.AppsLogin

Step 5 Verify on the NAM that you see the http session complete with a an HTTP 200 OK.

Step 6 On the ACE, add the http inspect policy to the multi match policy by issuing the following commands: **policy-map multi-match ORACLE_TCP_TRAFFIC class ORACLE_L4 inspect http policy DEEP**

Step 7 From a branch client, Send an HTTP request to http://wwwin-oefin.gslb.dcap.com:8000/OA_HTML/AppsLocalLogin.jsp?requestUrl=APPSHOMEPAGE&cancelUrl=http%3A%2F%2Fwwwin-oefin.gslb.dcap.com%3A8000%2Foa_servlets%2Foracle.apps.fnd.sso.AppsLogin

Step 8 On the primary CAT (dca-agg-1-ace-1), use the NAM to start a packet capture on the ACE's internal portchannel.

Step 9 Verify on the NAM that you are not seeing an HTTP 200 returned from the client due to the fact that the inspect policy in now in place and denying the content.

Step 10 On the ACE, display the HTTP inspect statitics by issuing the**show stats inspect** command:

Step 11 Verify that you are seeing the TCP option 0x21 being set by the WAAS as this traffic is traversing the WAE Vlan.

Step 12 On the ACE, change the http inspect policy by issuing the following commands on the ACE: **Make the following change on the ACE: class-map type http inspect match-all DEEP no match header length request range 129 255**

Step 13 Make the following configuration changes on ACE: **class-map type http inspect match-all DEEP 2 match request-method rfc post policy-map type inspect http all-match DEEP class DEEP policy-map multi-match ORACLE_TCP_TRAFFIC class ORACLE_L4 loadbalance vip inservice loadbalance policy GO_TO_WAE_FARM loadbalance vip icmp-reply inspect http policy DEEP**

Step 14 From a branch client, Send an HTTP request to: http://wwwin-oefin.gslb.dcap.com:8000/OA_HTML/AppsLocalLogin.jsp?requestUrl=APPSHOMEPAGE&cancelUrl=http%3A%2F%2Fwwwin-oefin.gslb.dcap.com%3A8000%2Foa_servlets%2Foracle.apps.fnd.sso.AppsLogin (sysadmin/sysadmin)

Step 15 On the Branch Client, set the MTU on the client to 576 using the setMTU tool or make a change to the registry.

Step 16 From the same branch client, Send an HTTP request to: http://wwwin-oefin.gslb.dcap.com:8000/OA_HTML/AppsLocalLogin.jsp?requestUrl=APPSHOMEPAGE&cancelUrl=http%3A%2F%2Fwwwin-oefin.gslb.dcap.com%3A8000%2Foa_servlets%2Foracle.apps.fnd.sso.AppsLogin (sysadmin/sysadmin)

Step 17 Verify the http inspect stats are increasing for total drop connections but issuing the **show stats inspect http** command.

Step 18 Stop background scripts to collect final status of network devices and analyze for error.

Step 19 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the ACE to property permit and deny http requests based on the client requests.
- We expect the ACE to send a TCP RESET to the client when REST is enabled on the HTTP inspect policy even when the client MTU is adjusted down to 576 bytes simulating a slow or dial-up user.
- We expect no CPU or memory problems.

Results

HTTP Inspection passed with exception. The following exceptions were noted: CSCsl66036.

Traffic Handling

The tests in this section look at how the ACE module advertises itself (via Virtual IPs) for incoming traffic and how it handles that incoming traffic.

Route Health Injection

To allow the ACE to advertise the IP address of the virtual server as the host route, use the loadbalance vip advertise command. This function is used with route health injection (RHI) to allow the ACE to advertise the availability of a VIP address throughout the network.

This section contains the following topics:

- ACE Oracle RHI, page 7-23

ACE Oracle RHI

The ACE can advertise the IP address of the virtual server as a static host route injected into the local MSFC. This function is commonly used with OSPF or a similar protocol to allow the ACE to advertise the availability of a VIP address throughout the network. RHI is used primarily in environments where the VIP is in a subnet that is not directly attached to the MSFC. This test verified that the host route was properly injected or removed in various conditions, probe failure, admin down, ft switchover, etc.

Relevant ACE Configuration:

```
probe http ORACLE_FORCED_FAIL
  port 8000
  interval 2
  faildetect 1
  passdetect interval 2
  credentials sysadmin sysadmin
  request method get url /does_not_exist.html
  expect status 200 200
probe http ORACLE_WEB_PAGE_CHECK_IND
  port 8000
```

```
  interval 2
  faildetect 1
  passdetect interval 2
  credentials sysadmin sysadmin
  request method get url /oa_servlets/AppsLogin
  expect status 200 200
  expect status 302 302
rserver host ORAAPP01
  description ORACLE_APP01
  ip address 101.1.33.16
  inservice
rserver host ORAAPP02
  description ORACLE_APP02
  ip address 101.1.33.5
  inservice
rserver host ORAAPP03
  description ORACLE_APP03
  ip address 101.1.33.47
  inservice
rserver redirect REDIRECT_80_TO_8000
  description 302 REDIRECT TO PORT 80
  webhost-redirection http://wwwin-oefin.gslb.dcap.com:8000 302
  inservice
serverfarm host ORAAPP_CINSERT_IND
  description ORACLE WEB AND DB SERVERS
  failaction purge
  probe ORACLE_WEB_PAGE_CHECK_IND
  rserver ORAAPP01
    inservice
  rserver ORAAPP02
    inservice
  rserver ORAAPP03
    inservice
serverfarm redirect REDIRECT_PORT_80_TO_PORT_8000_IND
  description REDIRECT CLIENTS FROM 80 To 8000
  failaction purge
  rserver REDIRECT_80_TO_8000
    inservice
sticky http-cookie ACE_COOKIE ORAAPP_CINSERT_IND
  cookie insert browser-expire
  replicate sticky
  serverfarm ORAAPP_CINSERT_IND
sticky http-cookie ACE_COOKIE ORAAPP_CINSERT2_IND
  cookie insert browser-expire
  replicate sticky
  serverfarm ORAAPP_CINSERT_IND
class-map type http loadbalance match-all ONLY_VALID_HTTP_FOR_REDIRCT_CANDIDATE
  2 match http header Host header-value "wwwin-oefin.gslb.dcap.com"
  3 match http url / method GET
class-map match-any ORACLE_L4_NO_WAAS_CINSERT2_IND
  2 match virtual-address 101.1.33.63 tcp eq 8000
class-map match-any ORACLE_L4_NO_WAAS_CINSERT_IND
  2 match virtual-address 101.1.33.62 tcp eq 8000
class-map match-all REDIRECT_VIP_L4_NO_WAAS_CINSERT2_IND
  2 match virtual-address 101.1.33.63 tcp eq www
class-map match-all REDIRECT_VIP_L4_NO_WAAS_CINSERT_IND
  2 match virtual-address 101.1.33.62 tcp eq www
class-map match-any ORACLE_RHI_90_IND
  2 match virtual-address 101.1.33.90 tcp eq 8000
class-map match-any ORACLE_RHI_91_IND
  2 match virtual-address 101.1.33.91 tcp eq 8000
class-map match-any ORACLE_RHI_92_IND
  2 match virtual-address 101.1.33.92 tcp eq 8000
...
```

```
...
...
class-map match-any ORACLE_RHI_148_IND
  2 match virtual-address 101.1.33.148 tcp eq 8000
class-map match-any ORACLE_RHI_149_IND
  2 match virtual-address 101.1.33.149 tcp eq 8000
class-map match-any ORACLE_RHI_150_IND
  2 match virtual-address 101.1.33.150 tcp eq 8000
policy-map type loadbalance first-match ORAAPP_SERVERS_CINSERT2_IND
  class class-default
    sticky-serverfarm ORAAPP_CINSERT2_IND
policy-map type loadbalance first-match ORAAPP_SERVERS_CINSERT_IND
  class L7_ORACLE
    sticky-serverfarm ORAAPP_CINSERT_IND
policy-map type loadbalance first-match REDIRECT_TCP:80_TO_TCP:8000_2_IND
  class class-default
    serverfarm REDIRECT_PORT_80_TO_PORT_8000_IND
policy-map type loadbalance first-match REDIRECT_TCP:80_TO_TCP:8000_IND
  class ONLY_VALID_HTTP_FOR_REDIRCT_CANDIDATE
    serverfarm REDIRECT_PORT_80_TO_PORT_8000_IND
policy-map type loadbalance first-match ORAAPP_RHI_90-150_IND
  class class-default
    serverfarm ORAAPP_CINSERT_IND
policy-map type inspect http all-match INSPECT_GOOD_HTTP
  class INSPECT_HTTP_GOOD
    permit
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
  class ORACLE_L4_NO_WAAS_CINSERT_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_SERVERS_CINSERT_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    inspect http policy INSPECT_GOOD_HTTP url-logging
  class REDIRECT_VIP_L4_NO_WAAS_CINSERT_IND
    loadbalance vip inservice
    loadbalance policy REDIRECT_TCP:80_TO_TCP:8000_IND
    loadbalance vip icmp-reply active
    inspect http policy INSPECT_GOOD_HTTP url-logging
  class ORACLE_L4_NO_WAAS_CINSERT2_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_SERVERS_CINSERT2_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
  class REDIRECT_VIP_L4_NO_WAAS_CINSERT2_IND
    loadbalance vip inservice
    loadbalance policy REDIRECT_TCP:80_TO_TCP:8000_2_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
policy-map multi-match ORACLE_RHI_90-150_IND
  class ORACLE_RHI_90_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_RHI_90-150_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
  class ORACLE_RHI_91_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_RHI_90-150_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
  class ORACLE_RHI_92_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_RHI_90-150_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
```

```
...
...
...
  class ORACLE_RHI_148_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_RHI_90-150_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
  class ORACLE_RHI_149_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_RHI_90-150_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
  class ORACLE_RHI_150_IND
    loadbalance vip inservice
    loadbalance policy ORAAPP_RHI_90-150_IND
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
interface vlan 1133
  description SERVER_VLAN
  bridge-group 10
  ip options allow
  mtu 2000
  no normalization
  fragment min-mtu 68
  no icmp-guard
  access-group input BPDU-ALLOW
  access-group input anyone
  access-group output anyone
  service-policy input NAT_POLICY
  service-policy input REMOTE-MGNT
  no shutdown
interface vlan 2132
  description CLIENT_VLAN
  bridge-group 10
  ip options allow
  mtu 2000
  no normalization
  fragment min-mtu 68
  no icmp-guard
  access-group input BPDU-ALLOW
  access-group input anyone
  access-group output anyone
  service-policy input ORACLE_TCP_TRAFFIC
  service-policy input ORACLE_TCP_TRAFFIC_IND
  service-policy input ORACLE_RHI_90-150_IND
  no shutdown
interface bvi 10
  ip address 101.1.33.85 255.255.255.0
  alias 101.1.33.87 255.255.255.0
  peer ip address 101.1.33.86 255.255.255.0
  description client-side L3
  no shutdown
```

Test Procedure

The procedure used to perform the ACE Oracle RHI test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the primary ACE (dca-agg-1-ace-1), c2 context, the following vservers are configured to advertise the route as shown below:

```
class: ORACLE_L4_NO_WAAS_CINSERT_IND         (Advertise:ENABLED-WHEN-ACTIVE)
class: REDIRECT_VIP_L4_NO_WAAS_CINSERT_IND   (Advertise:DISABLED)
class: ORACLE_L4_NO_WAAS_CINSERT2_IND        (Advertise:ENABLED-WHEN-ACTIVE)
class: REDIRECT_VIP_L4_NO_WAAS_CINSERT2_IND  (Advertise:ENABLED-WHEN-ACTIVE)
```

Verify that these vservers are inservice and configured to advertise as shown by issuing the

Step 3 On the primary 6k verify that VIPs 101.1.33.62 and 101.1.33.63 are injected as a static route with a metric of 77. Verify this by issuing the **show ip route static** command.

Step 4 Take vserver 101.1.33.62 out-of-service by adding a bad probe to serverfarm ORAAPP_CINSERT_IND. Issuing the following commands:

```
config
serverfarm ORAAPP_CINSERT_IND
probe ORACLE_FORCED_FAIL
end
```

Step 5 Verify that the serverfarm has failed and class maps, ORACLE_L4_NO_WAAS_CINSERT_IND, ORACLE_L4_NO_WAAS_CINSERT2_IND are no longer inservice. Issue the following commands:

```
show serverfarm ORAAPP_CINSERT_IND detail
show service-policy ORACLE_TCP_TRAFFIC_IND detail
```

Step 6 Verify on the primary 6K that VIP 101.1.33.62 has been removed and that 101.1.33.63 is still present by issuing the **show ip route static** command.

Step 7 VIP 101.1.33.63 is still advertised because class map REDIRECT_VIP_L4_NO_WAAS_CINSERT2_IND uses the same IP, is configured to advertise, and is still active. Disable its serverfarm by issuing the following commands:

```
config
serverfarm redirect REDIRECT_PORT_80_TO_PORT_8000_IND
rserver REDIRECT_80_TO_8000
no inservice
end
```

Step 8 Verify on the primary 6K that VIP 101.1.33.63 has been removed by issuing the **show ip route static** command.

Step 9 These VIPs are configured to advertise only when active. Change the config for VIP 101.1.33.62 to always advertise, regardless of state. Issue the following commands:

```
config
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
class ORACLE_L4_NO_WAAS_CINSERT_IND
loadbalance vip advertise
end
```

Step 10 On the primary 6k verify that VIP 101.1.33.62 is injected by issuing the **show ip route static** command.

Step 11 By default the VIP is inserted as a static route with a metric of 77. Change the config for VIP 101.1.33.62 to use a metric of 254. Issue the following commands:

```
config
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
class ORACLE_L4_NO_WAAS_CINSERT_IND
loadbalance vip advertise metric 254
end
```

Step 12 On the primary 6k verify that VIP 101.1.33.62 is injected with a metric of 254 by issuing the **show ip route static** command.

Step 13 Restore the config back to its original setting by issuing the following commands:

```
config
serverfarm ORAAPP_CINSERT_IND
no probe ORACLE_FORCED_FAIL
serverfarm redirect REDIRECT_PORT_80_TO_PORT_8000_IND
rserver REDIRECT_80_TO_8000
inservice
exit
exit
policy-map multi-match ORACLE_TCP_TRAFFIC_IND
class ORACLE_L4_NO_WAAS_CINSERT_IND
no loadbalance vip advertise metric 254
loadbalance vip advertise active
end
```

Step 14 The active ACE when injecting a VIP into the routing table will use the alias address of the VLAN or BVI, if available. If an alias address is not used it will use the circuit IP. In this test the ACE will use the alias IP configured for VLANs 2132/1133 (BVI 10), so it will use 101.1.33.87. Verify this next hop address and count the number of injected routes addressed as 101.1.33.x by issuing the **show ip route static**.

Step 15 Force a fault tolerant failover of the c2 context by issuing the following commands:

```
changeto Admin
ft switchover 1
```

Step 16 Verify that the primary ACE is in a standby_hot state and that all of the VIPs counted prior to FT switchover are injected as static routes on the secondary 6K (dca-agg-2). Issue the following commands:

```
show ft group 1 detail (dca-agg-1-ace-1)
show ip route static   (dca-agg-2)
```

Step 17 On a Linux client send a burst of traffic and verify through the CLI output that these requests were successful.

Step 18 Force a fault tolerant failover of the c2 context by issuing the following commands:

```
changeto Admin
ft switchover 1
```

Step 19 Verify that the secondary ACE is in a standby_hot state and that all of the VIPs counted prior to FT switchover are injected as static routes on the primary 6K (dca-agg-1). Issue the following commands:

```
show ft group 1 detail (dca-agg-1-ace-1)
show ip route static   (dca-agg-1)
```

Step 20 On a Linux client send a burst of traffic and verify through the CLI output that these requests were successful.

Step 21 Repeat the failover steps for two more complete sequences, failover to standby and back again. Verify that the proper number of routes are injected and the traffic is successful.

Step 22 Stop background scripts to collect final status of network devices and analyze for error.

Step 23 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect host routes to be injected when the vservers are active.
- We expect the host routes to be removed when the vserver becomes out of service.
- We expect the appropriate route and gateway to be injected during a redundancy failure.
- We expect host routes to be injected regardless of the vserver state when configured accordingly.
- We expect no CPU or memory problems.

Results

ACE Oracle RHI passed.

CHAPTER 8

IDSM IPS

Tests in the "IDSM IPS" section on page 8-1 focus on the functionality of the IDSM service module, operating in the DCAP environment. Results for additional tests run by the Safe Harbor team against the software version tested in DCAP 4.0 are available in the DCAP 4.0 Appendix.

Refer to the following chapters for respective Layer 4-7 services testing:

- CSM—Based Integrated Switch Bundle "Layer 4-7 CSM" section on page 2-1
- ACE—Based Integrated Switch Bundle "Layer 4-7 ACE" section on page 3-1
- CSM—Based Service Chassis Bundle "Layer 4-7 Services Switch" section on page 4-1
- Application Control Engine (ACE) "ACE" section on page 5-1

Test Results Summary

Table 8-1 on page 8-2 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 8-1 on page 8-2 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 8-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Failure scenarios, page 8-3	n/a	1. IDSM Module Reload 2. IDSM Module removal	
IDSM Baseline, page 8-5	n/a	1. Baseline Throughput	
Threat detection under load, page 8-6	n/a	1. Threat 1104 IP Localhost Source Spoof 2. Threat 1108 IP Packet with Protocol 11 3. Threat 3041 TCP SYNFIN Packet 4. Threat 4003 Nmap UDP Port Sweep	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Failure scenarios

This set of tests verify the ability of the IDSM to service traffic under certain failure scenarios. Failure Tests verify hardware failures of the modules under test and their ability to service traffic under failure situations.

This section contains the following topics:

IDSM Module Reload

This test case tests the failure of the IDSM does not effect traffic when designed with a loop failure mechanism. The IDSM-2 itself doesnt have a failover mechanism build in. This means a method to keep traffic flowing while the IDSM is down needs to be implemented. This method is documented in the Design guide for all IPS systems. Even though the IDSM is a cat6k module it behaves similar to the IPS Appliance. To allow an uninterruptible traffic flow, we have to use a physical loopback cable. This solution allows us to have a redundant path from Vlan 1140 to Vlan 4001 with STP assuring a loop free topology.

Test Procedure

The procedure used to perform the IDSM Module Reload test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Check stp for vlan 4001. Verify that the IDSM port (gi4/d2) is in the FWD state and the port gi12/10 in the BLK state. The IDSM uses vlan 1140 for the outside vlan and vlan 4001 for the inside vlan. The Supervisor identifies the inbound IDSM interface with Gi4/d1 and the outbound interface with Gi4/d2. This step verifies that STP blocks the correct path and allows traffic to the IDSM-2. **sh spanning-tree vlan 4001**

Step 3 Run normal IXIA HTTP and FTP traffic through the IDSM To have a graphical output of the Traffic flow were using the IXIA IDSM-full.rxf to generate HTTP and FTP traffic. These diagrams are used later to verify that no traffic loss occurred during the module reset.

Step 4 Clear interface counters for the failover interfaces gi12/9 and 12/10 **clear counters**

Step 5 verify that no traffic is passing both interfaces **show interface gigabitethernet 12/9show interface gigabitethernet 12/10**

Step 6 fail the IDSM-2 The reset test includes a restart of the IDSM to interrupt the traffic path. Verify reset with the show module command. **hw-module module 4 resetshow module**

Step 7 Check the IXIA statistics for any traffic loss

Step 8 Verify that the IDSM port (gi4/d2) is removed from the STP output and that the failover port (gi12/10) changed it state from BLK to FWD **show spanning-tree vlan 4001**

Step 9 Verify that traffic is passing both interfaces now **show interface gigabitethernet 12/9show interface gigabitethernet 12/9**

Step 10 Wait for the IDSM to come back online **show module**

Step 11 verify that the gi4/d2 port comes back up and is in the FWD state for the vlan 4001 **show spanning-tree vlan 4001**

Step 12 Stop the ixia traffic, generate a detailed report and upload the report

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Traffic not to be continue via the loopback cable when there is a failure.

Results

IDSM Module Reload passed.

IDSM Module removal

This test case tests the failure of the IDSM does not effect traffic when designed with a loop failure mechanism. The IDSM-2 itself doesnt have a failover mechanism build in. This means a method to keep traffic flowing while the IDSM is down needs to be implemented. This method is documented in the Design guide for all IPS systems. Even though the IDSM is a cat6k module it behaves similar to the IPS Appliance. To allow an uninterruptible traffic flow, we have to use a physical loopback cable. This solution allows us to have a redundant path from Vlan 1140 to Vlan 4001 with STP assuring a loop free topology.

Test Procedure

The procedure used to perform the IDSM Module removal test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Check stp for vlan 4001. Verify that the IDSM port (gi4/d2) is in the FWD state and the port gi12/10 in the BLK state. The IDSM uses vlan 1140 for the outside vlan and vlan 4001 for the inside vlan. The Supervisor identifies the inbound IDSM interface with Gi4/d1 and the outbound interface with Gi4/d2. This step verifies that STP blocks the correct path and allows traffic to the IDSM-2. **sh spanning-tree vlan 4001**

Step 3 Run normal IXIA HTTP and FTP traffic through the IDSM To have a graphical output of the Traffic flow were using the IXIA IDSM-full.rxf to generate HTTP and FTP traffic. These diagrams are used later to verify that traffic loss during the module reset.

Step 4 Clear interface counters for the failover interfaces gi12/9 and 12/10 **clear counters**

Step 5 verify that no traffic is passing both interfaces **show interface gigabitethernet 12/9show interface gigabitethernet 12/10**

Step 6 fail the IDSM-2 For this test the IDSM must be physically removed from the chassis. **show module**

Step 7 Check the IXIA statistics for any traffic loss

Step 8 Verify that the IDSM port (gi4/d2) is removed from the STP output and that the failover port (gi12/10) changed it state from BLK to FWD **show spanning-tree vlan 4001**

Step 9 Verify that traffic is passing both interfaces now **show interface gigabitethernet 12/9show interface gigabitethernet 12/9**

Step 10 Insert the IDSM module and wait for it to come back online **show module**

Step 11 verify that the gi4/d2 port comes back up and is in the FWD state for the vlan 4001 **show spanning-tree vlan 4001**

Step 12 Stop the ixia traffic, generate a detailed report and upload the report

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Traffic not to be continue via the loopback cable when there is a failure.

Results

IDSM Module removal passed.

IDSM Baseline

The tests here verify the basic functionality of the IDSM in the test topology. Baseline tests verify network is in working order prior to starting testing and quantify steady state network performance.

This section contains the following topics:

Baseline Throughput

This test shows the throughput without IDSM in the traffic flow (module shutdown), IDSM in traffic flow with a turned off inspection engine and IDSM in traffic flow with inspection engine turned on. This test allows determining the behaviour of the traffic throughput with and without IDSM in the traffic path. the IDSM virtualSensor must be disabled prior to starting this test

Test Procedure

The procedure used to perform the Baseline Throughput test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Shutdown IDSM-2 module **no power enable module 4**

Step 3 Run normal IXIA HTTP traffic, use IDSM-http-full.rxf file To have a graphical output of the Traffic flow were using the IXIA IDSM-http-full.rxf to generate HTTP traffic. These diagrams are used later to verify the throughput baseline

Step 4 Wait 10 minutes and then power on the IDSM module **power enable module 4**

Step 5 When the module is back online check the Ixia statistics and make sure there is a reduction in the throughput.

Step 6 Wait 10 minutes and then enable the Inspection engine Use GUI to turn on inspection engine on IDSM The IDSM allows it to run traffic through it without having the inspection engine enabled. To check if the inspection engine is disabled follow these steps: Load IDM Go to configuration Select the menu point Virtual Sensors under Analysis Engine Edit inspection engine Assign Inline (Interface Pair of Gig 0/7 and Gig 0/8) Interface to the inspection engine

Step 7 Wait 10 minutes Check the IXIA statistics and upload the detailed report

Step 8 Repeat using FTP traffic

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect a smaller throughput when the IDSM module is place in the traffic path
- We expect an even smaller throughput when the inspection engine is turned on.

Results

Baseline Throughput passed.

Threat detection under load

Threat detection test cases look at the ability of the IDSM module to handle normal traffic under threat situations and correctly block threatening traffic whilst continuing to pass normal traffic.

This section contains the following topics:

Threat 1104 IP Localhost Source Spoof

This test verified that depending on the threat which was injected into the network, that the correct alarm was generated. HPING3 test tool loaded on to a client machine was used to generate the threat. This signature triggers when an IP packet with a address of 127.0.0.1 is detected. This is a local host IP address and should never be seen on the network This may be indicative of someone trying to take advantage of local host trust relationships to either gain access to or in some other way subvert a target machine.

Test Procedure

The procedure used to perform the Threat 1104 IP Localhost Source Spoof test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify all signatures are enabled using the IDM GUI and that the virtual sensor is enabled and assigned to the port pairs.

Step 3 Clear the service policy statistics on the ACE module with the following command **clear service-policy IXIA-IDSM-POLICY**

Step 4 Start background traffic using the IXIA. Direct the ixia traffic at the VIP IXIA-IDSM and run a throughput test for both HTTP and FTP traffic.

Step 5 Verify traffic is hitting the Service policy with the following command **show service-policy IXIA-IDSM-POLICY**

Step 6 After the Ixia traffic has started and ramped up has completed initiate the IP localhost scource spoof attack **hping send {ip(saddr=127.0.0.1,daddr=101.3.40.200,ttl=255)+tcp(sport=123,dport=80)}**

Step 7 Check using the IDM GUI that an alert message is seen in the Event Viewer to prove the IDSM has detected the threat evIdsAlert: eventId=1195588102275430240 vendor=Cisco severity=high originator: hostId: AGG1-IDSM appName: sensorApp appInstanceId: 566 time: December 14, 2007 11:43:50 AM UTC offset=0 timeZone=UTC signature: description=IP Localhost Source Spoof id=1104 version=S2 subsigId: 0 sigDetails: IP Localhost Source Spoof marsCategory: Penetrate/SpoofIdentity/TCPIP interfaceGroup: vs0 vlan: 0 participants: attacker: addr: 127.0.0.1 locality=OUT port: 123 target: addr: 101.3.40.200 locality=OUT port: 80 os: idSource=unknown type=unknown relevance=relevant actions: droppedPacket: true deniedFlow: true riskRatingValue: 100 targetValueRating=medium attackRelevanceRating=relevant threatRatingValue: 65 interface: ge0_7 protocol: tcp

Step 8 Stop the Ixia traffic and upload the detailed report

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the correct alarms to be generated when a threat is detected.
- We expect no CPU or memory problems.

Results

Threat 1104 IP Localhost Source Spoof passed.

Threat 1108 IP Packet with Protocol 11

This test verified that depending on the threat which was injected into the network, that the correct alarm was generated. HPING3 test tool loaded on to a client machine was used to generate the threat. This signature triggers when an IP packet with a address of 127.0.0.1 is detected. This is a local host IP address and should never be seen on the network This may be indicative of someone trying to take advantage of local host trust relationships to either gain access to or in some other way subvert a target machine.

Test Procedure

The procedure used to perform the Threat 1108 IP Packet with Protocol 11 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify all signatures are enabled using the IDM GUI and that the virtual sensor is enabled and assigned to the port pairs.

Step 3 Clear the service policy statistics on the ACE module with the following command **clear service-policy IXIA-IDSM-POLICY**

Step 4 Start background traffic using the IXIA. Direct the ixia traffic at the VIP IXIA-IDSM and run a throughput test for bothe HTTP and FTP traffic.

Step 5 Verify traffic is hitting the Service policy with the following command **show service-policy IXIA-IDSM-POLICY**

Step 6 After the Ixia traffic has started and ramped up has completed initiate the IP packet with protocol 11 threat. **hping send {ip(saddr=102.1.1.200,daddr=101.3.40.200,ttl=255,proto=11)+tcp(sport=123,dport=80)}**

Step 7 Check using the IDM GUI that an alert message is seen in the Event Viewer to prove the IDSM has detected the threat evIdsAlert: eventId=1195588102275430195 vendor=Cisco severity=high originator: hostId: AGG1-IDSM appName: sensorApp appInstanceId: 566 time: December 14, 2007 11:42:46 AM UTC offset=0 timeZone=UTC signature: description=IP Packet with Proto 11 id=1108 version=S27 subsigId: 0 sigDetails: IP Proto 11 marsCategory: Penetrate/Backdoor/Trojan/Connect interfaceGroup: vs0 vlan: 0 participants: attacker: addr: 102.1.1.200 locality=OUT target: addr: 101.3.40.200 locality=OUT os: idSource=unknown type=unknown relevance=relevant actions: droppedPacket: true riskRatingValue: 100 targetValueRating=medium attackRelevanceRating=relevant threatRatingValue: 65 interface: ge0_7 protocol: IP protocol 11

Step 8 Stop the Ixia traffic and upload the detailed report

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the correct alarms to be generated when a threat is detected.
- We expect no CPU or memory problems.

Results

Threat 1108 IP Packet with Protocol 11 passed.

Threat 3041 TCP SYNFIN Packet

This test verified that depending on the threat which was injected into the network, that the correct alarm was generated. HPING3 test tool loaded on to a client machine was used to generate the threat. Triggers when a single TCP packet with the SYN and FIN flags are set and is sent to a specific host. This is indicative that a reconnaissance sweep of your network may be in progress. The use of this type of packet indicates an attempt to conceal the sweep. This may be the prelude to a more serious attack. This should never occur in legitimate traffic. The source of this packet should be shunned.

Test Procedure

The procedure used to perform the Threat 3041 TCP SYNFIN Packet test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify all signatures are enabled using the IDM GUI and that the virtual sensor is enabled and assigned to the port pairs.

Step 3 Clear the service policy statistics on the ACE module with the following command **clear service-policy IXIA-IDSM-POLICY**

Step 4 Start background traffic using the IXIA. Direct the ixia traffic at the VIP IXIA-IDSM and run a throughput test for bothe HTTP and FTP traffic.

Step 5 Verify traffic is hitting the Service policy with the following command **show service-policy IXIA-IDSM-POLICY**

Step 6 After the Ixia traffic has started and ramped up has completed initiate the TCP syn/fin attack. **hping send {ip(saddr=102.1.1.200,daddr=101.3.40.200,ttl=255,proto=6)+tcp(sport=123,dport=80,flags=sf)}**

Step 7 Check using the IDM GUI that an alert message is seen in the Event Viewer to prove the IDSM has detected the threat evIdsAlert: eventId=1195588102275429546 vendor=Cisco severity=high originator: hostId: AGG1-IDSM appName: sensorApp appInstanceId: 566 time: December 14, 2007 11:24:56 AM UTC offset=0 timeZone=UTC signature: description=TCP SYN/FIN Packet id=3041 version=S2 subsigId: 0 marsCategory: Probe/Host/Stealth interfaceGroup: vs0 vlan: 0 participants: attacker: addr: 102.1.1.200 locality=OUT port: 123 target: addr: 101.3.40.200 locality=OUT port: 80 os: idSource=unknown type=unknown relevance=relevant actions: droppedPacket: true deniedFlow: true riskRatingValue: 100 targetValueRating=medium attackRelevanceRating=relevant threatRatingValue: 65 interface: ge0_7 protocol: tcp

Step 8 Stop the Ixia traffic and upload the detailed report

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the correct alarms to be generated when a threat is detected.
- We expect no CPU or memory problems.

Results

Threat 3041 TCP SYNFIN Packet passed.

Threat 4003 Nmap UDP Port Sweep

This test verified that depending on the threat which was injected into the network, that the correct alarm was generated. NMAP test tool loaded on to a client machine was used to generate the threat. An open port scan threat triggers when a series of UDP connections to a number of different privileged ports (having port number less than 1024) on a specific host have been initiated. This is indicative that a reconnaissance sweep of your network may be in progress. This may be the prelude to a more serious attack.

Test Procedure

The procedure used to perform the Threat 4003 Nmap UDP Port Sweep test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify all signatures are enabled using the IDM GUI and that the virtual sensor is enabled and assigned to the port pairs.

Step 3 Clear the service policy statistics on the ACE module with the following command **clear service-policy IXIA-IDSM-POLICY**

Step 4 Start background traffic using the IXIA. Direct the ixia traffic at the VIP IXIA-IDSM and run a throughput test for both HTTP and FTP traffic.

Step 5 Verify traffic is hitting the Service policy with the following command **show service-policy IXIA-IDSM-POLICY**

Step 6 After the Ixia traffic has started and ramped up has completed initiate the NMAP open port scan using the command **nmap -sU 101.3.40.200**

Step 7 Check using the IDM GUI that an alert message is seen in the Event Viewer to prove the IDSM has detected the threat evIdsAlert: eventId=1195588102275428060 vendor=Cisco severity=high originator: hostId: AGG1-IDSM appName: sensorApp appInstanceId: 566 time: December 14, 2007 10:46:40 AM UTC offset=0 timeZone=UTC signature: description=Nmap UDP Port Sweep id=4003 version=S160 subsigId: 0 sigDetails: Nmap UDP port sweep marsCategory: Probe/FromScanner interfaceGroup: vs0 vlan: 0 participants: attacker: addr: 102.1.1.252 locality=OUT port: 59672 target: addr: 101.3.40.200 locality=OUT port: 792 port: 54 port: 270 port: 367 port: 571 port: 250 port: 712 port: 806 port: 117 os: idSource=unknown type=unknown relevance=relevant actions: droppedPacket: true riskRatingValue: 75 targetValueRating=medium attackRelevanceRating=relevant threatRatingValue: 40 interface: ge0_7 protocol: udp

Step 8 Stop the Ixia traffic and upload the detailed report

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the correct alarms to be generated when a threat is detected.
- We expect no CPU or memory problems.

Results

Threat 4003 Nmap UDP Port Sweep passed.

CHAPTER 9

Storage Area Networking (SAN)

DCAP SAN testing incorporates Cisco MDS fabric director products and design guides, industry best practices, and storage vendor implementation guidelines to provide a SAN infrastructure that is representative of the typical enterprise data center environment. The centerpiece of the topology configuration is the Cisco MDS 9500 multiprotocol SAN director running SAN-OS version 3.1(3a).

SAN Topology

The topology provides redundant fiber channel connectivity for Linux and Windows hosts using QLogic and Emulex host bus adaptors to three different types of fiber channel enterprise storage arrays, namely the EMC DMX3, Network Appliance FAS6070, and Hewlett Packard XP10000. The topology also provides redundant fiber channel connectivity for synchronous storage replication and fiber channel over IP connectivity for asynchronous storage replication. Delay simulators allow modeling of a redundant data center environment for disaster recovery and business continuance testing. The topology is designed to use actual hosts and applications to generate test traffic to model actual customer environments as close as possible.

Figure 9-1 depicts the entire SAN topology, including MDS switches, end devices, and test devices. The MDS switches are mostly MDS9513s with these components:

- Redundant version 2 supervisors for high availability and nondisruptive upgrade capability.
- FC modules with 12, 24, and 48 4-Gbps capable ports used for host and storage connectivity as well as interswitch links.
- Storage Service Modules with 32 2-Gbps capable FC ports used for testing replication with FC write acceleration.
- 14+2 modules with 14 2-Gbps capable FC ports and 2 Gigabit Ethernet ports for FCIP.

The MDS switches provide the following services:

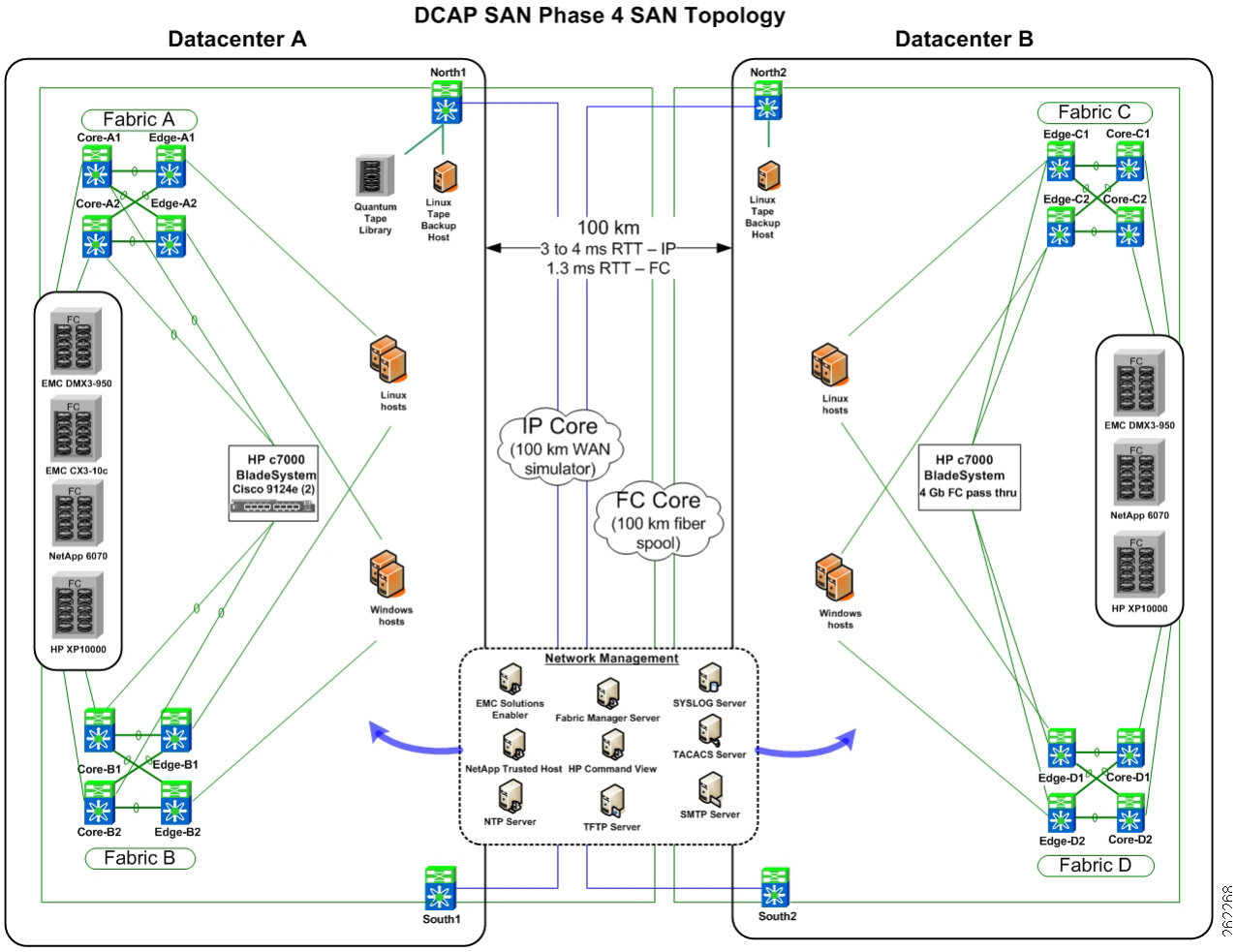
- Dual, redundant FC fabrics for connecting hosts to storage
- Dual FC fabrics for synchronous replication over a transit core consisting of two 100-km fiber spools and CWDM optics
- Dual FCIP fabrics for asynchronous storage replication over a transit core consisting of Gigabit Ethernet links.

All FC fabrics are implemented using Cisco's VSAN technology. All interswitch links belong to port channels for redundancy.

The end devices include hosts, storage arrays, and a tape library. The hosts are running Linux and Windows and have either 4-Gbps Qlogic or 2-Gbps Emulex HBAs providing two redundant paths to storage devices. The storage arrays include EMC DMX3 and CX3, Hewlett Packard XP10000s, and Network Appliance FAS6070s. The tape library is a Quantum (formerly ADIC) Scalar i500 with two IBM LTO3 UDS3 tape drives with 4-Gbps FC interfaces capable of 80 MB/sec uncompressed throughput and 400 GB uncompressed capacity per tape. In general, hosts connect to edge switches and storage arrays connect to core switches. The only exceptions are the tape hosts, tape library, and drives, which use the transit core switches. No replication testing was done on EMC CX3 arrays in this phase.

The test devices include an Agilent SAN tester with three 4-port N2X 4 Gbps blades and two Linux-based Gigabit Ethernet delay generators. The test devices are used in favor of end devices only when end devices are unsuitable or incapable of supplying the required test conditions.

Figure 9-1 DCAP SAN Test Topology Overview



Transport Core

Figure 9-2 shows the infrastructure used by the storage arrays to replicate data between data centers. The FC transit core consists of a pair of Cisco DS-CWDM-OADMA optical add/drop multiplexers which provide four bidirectional circuits over four separate wavelengths through a pair of 100 km optical

fibers. Two circuits comprise the north transit fabric and the other two the south transit fabric. The circuits in each fabric support native FC traffic for synchronous replication. The IP transit core consists of connections to Catalyst 6500 access switches which are part of the data center LAN as well as WAN edge switches. All connectivity is with Gigabit Ethernet. The WAN switches are connected to each other through Linux hosts running network emulation (netem) software which allows latency generation for distance simulation and bandwidth limiting. The IP transit core supports FCIP traffic for asynchronous replication.

Figure 9-3 shows the virtual SANs (VSANs) used to implement the dual fabric configuration for host to storage connectivity and storage to storage replication. Separate VSANs facilitate logical isolation of traffic by storage frame vendor for some tests and allow tuning FC and FCIP protocols independently.

Figure 9-2 DCAP SAN Test Topology Transit Core Detail

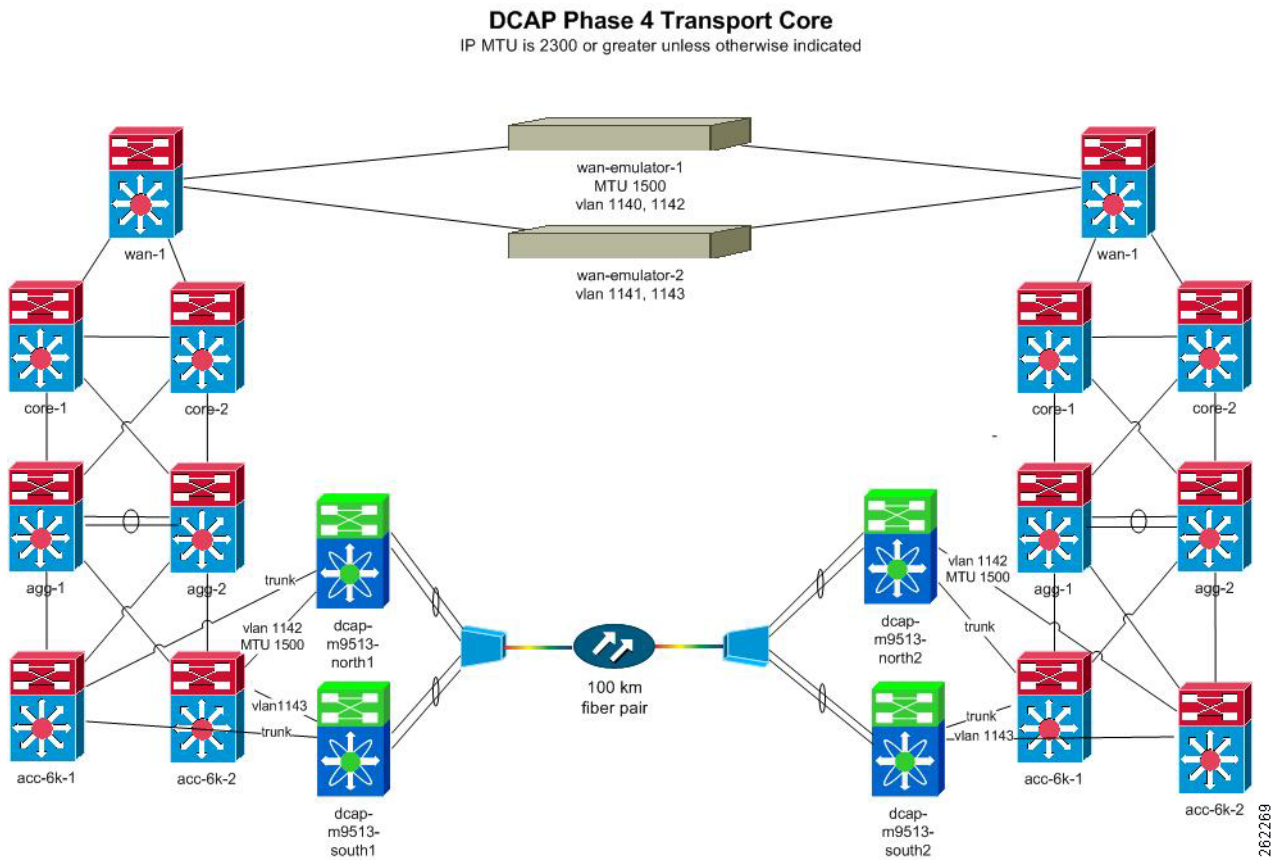


Figure 9-4 shows the host and storage ports in Fabric A.

Figure 9-5 shows the host and storage ports in Fabric B.

Figure 9-6 shows the host and storage ports in Fabric C.

Figure 9-7 shows the host and storage ports in Fabric D.

Figure 9-8 shows the storage replication ports in the North Transit Fabric.

Figure 9-9 shows the storage replication ports in the South Transit Fabric.

Figure 9-3 DCAP SAN VSAN Data Flow

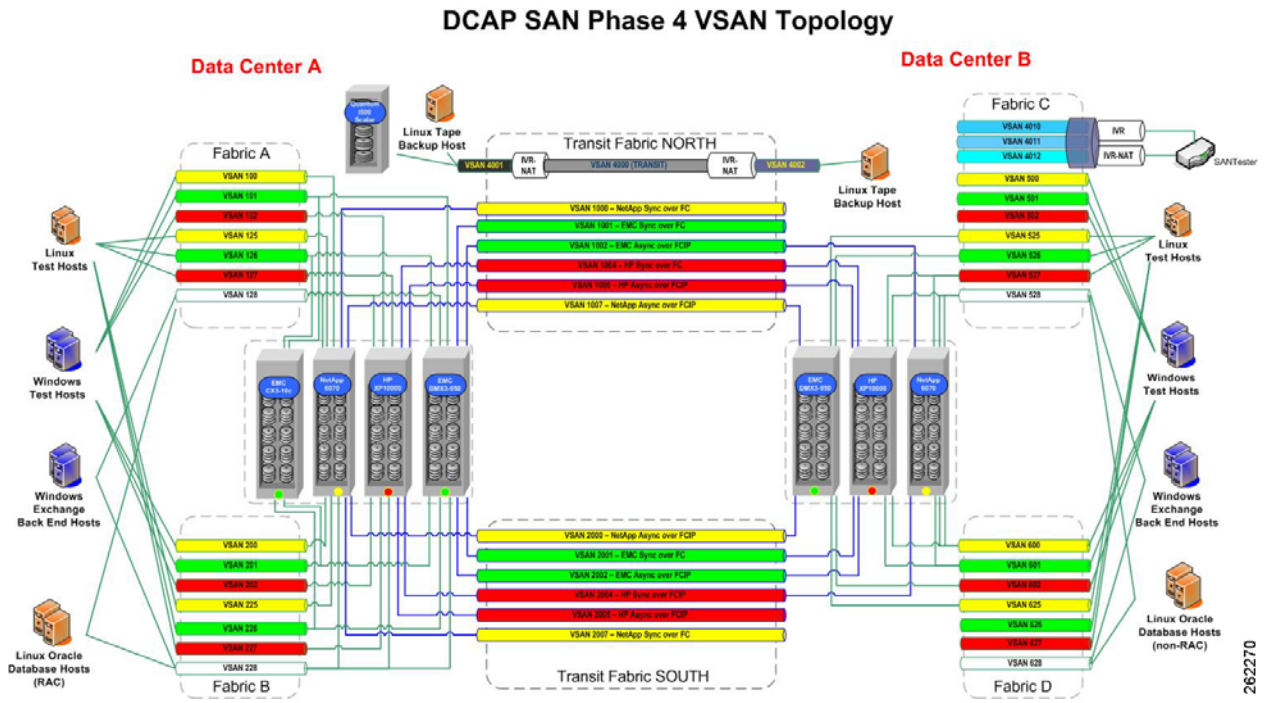


Figure 9-4 Fabric Manager Topology Map for Fabric A

:

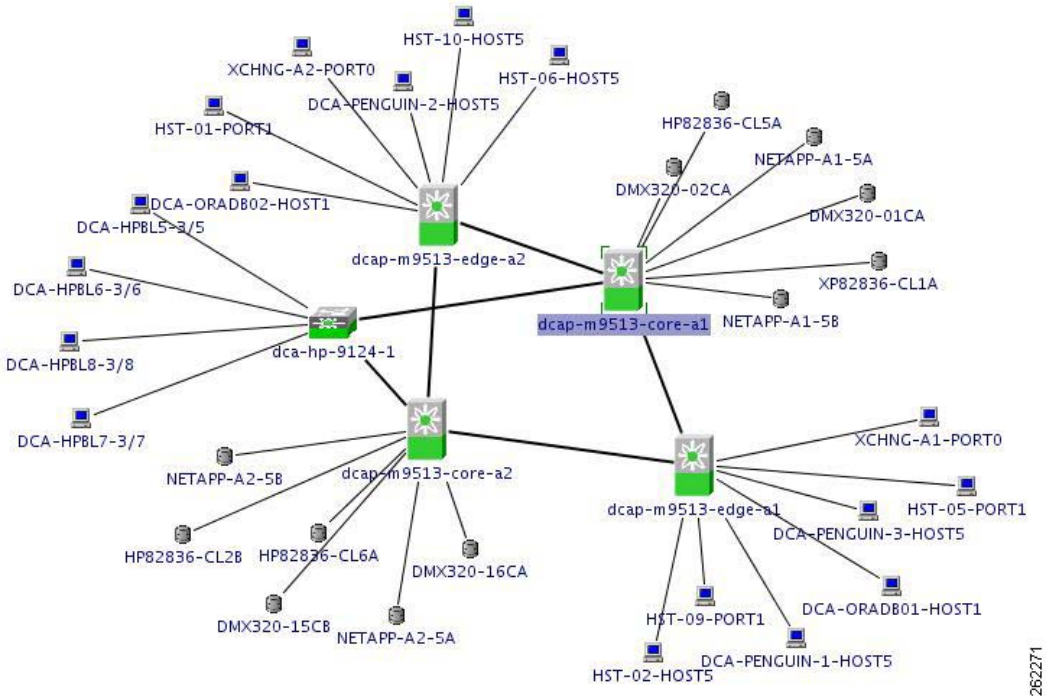


Figure 9-5 Fabric Manager Topology Map for Fabric B

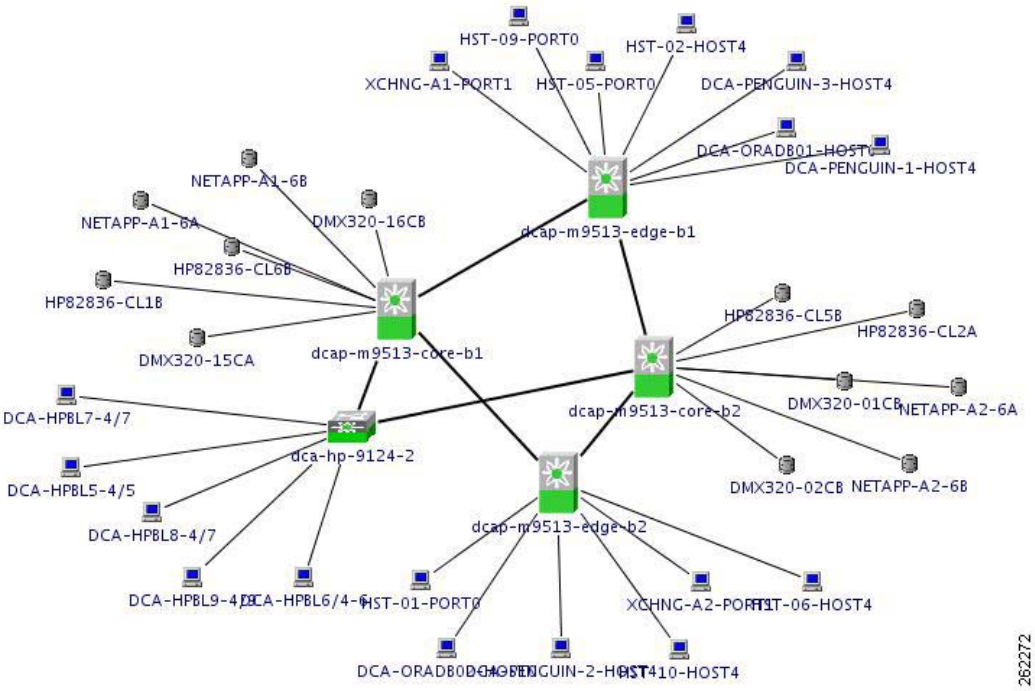


Figure 9-6 Fabric Manager Topology Map for Fabric C

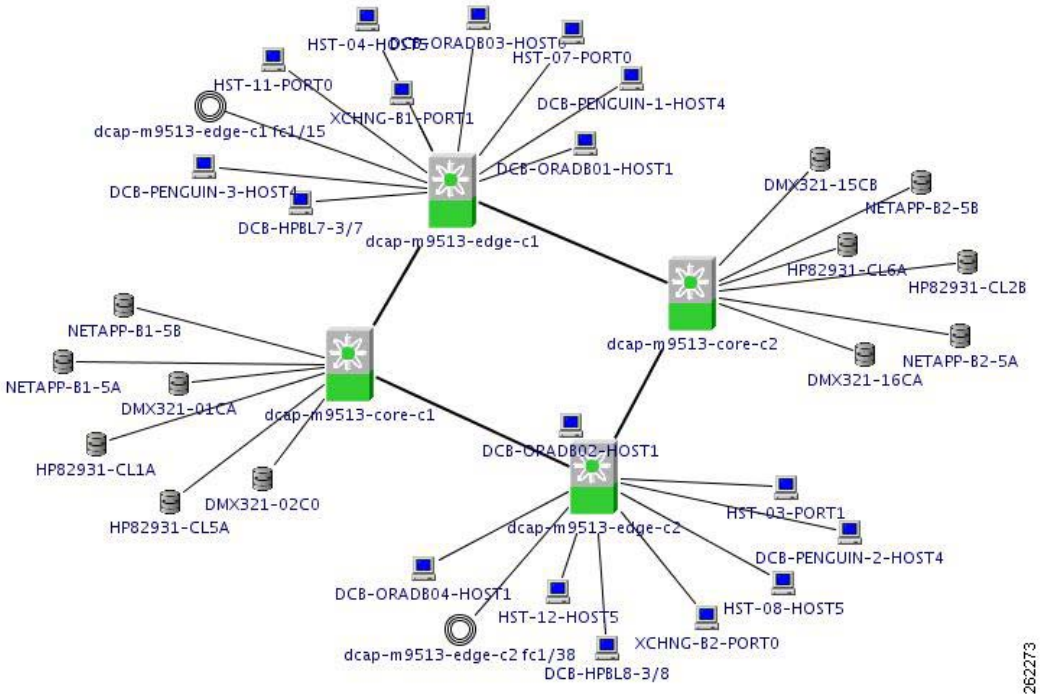


Figure 9-7 Fabric Manager Topology Map for Fabric D

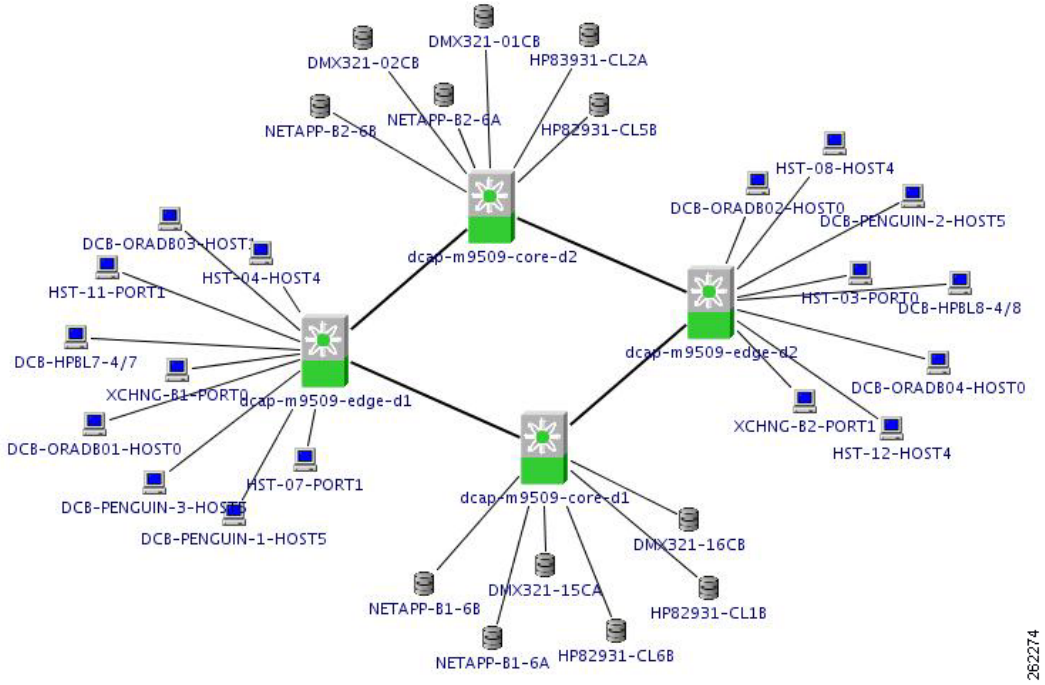


Figure 9-8 Fabric Manager Topology Map for North Transit Fabric

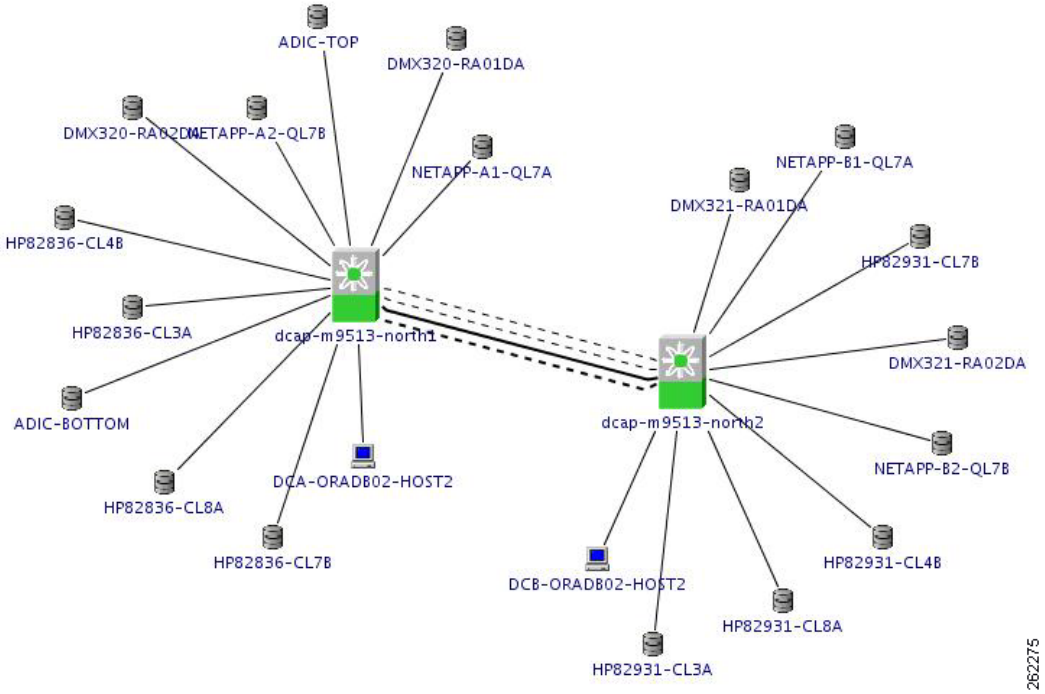


Figure 9-9 Fabric Manager Topology Map for South Transit Fabric

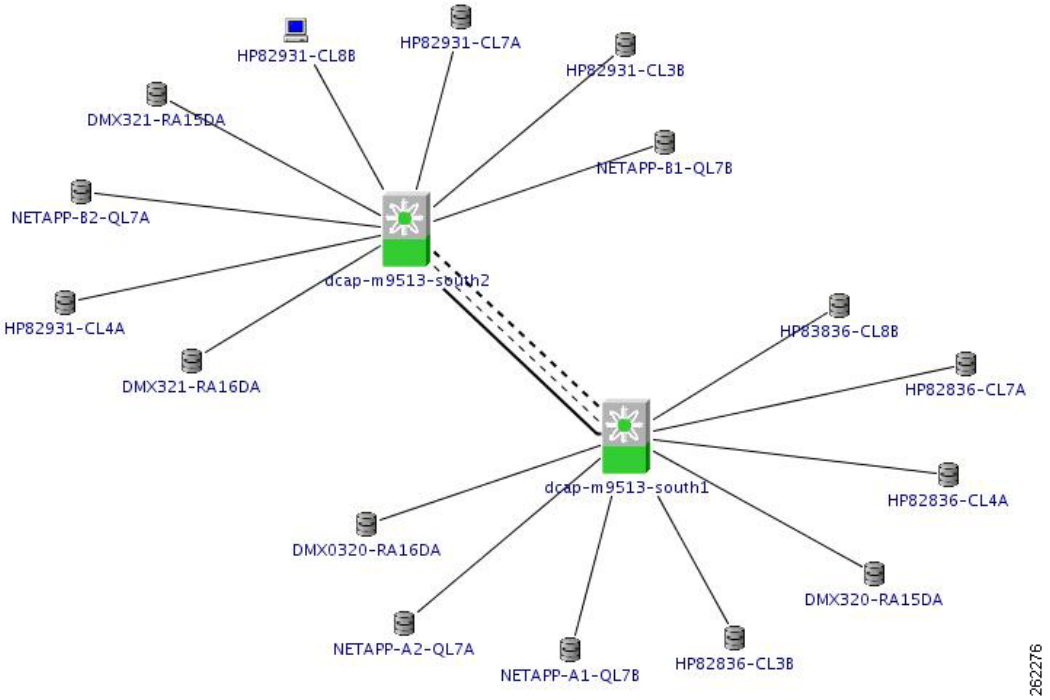


Figure 9-10 summarizes synchronous FC replication results. Note that write acceleration was not tested at the 0 km distance. Also note that Network Appliance SnapMirror doesn't benefit from write acceleration due to use of the IPFC and FC-VI protocols rather than FCP.

Figure 9-10 Synchronous FC Replication Result Summary

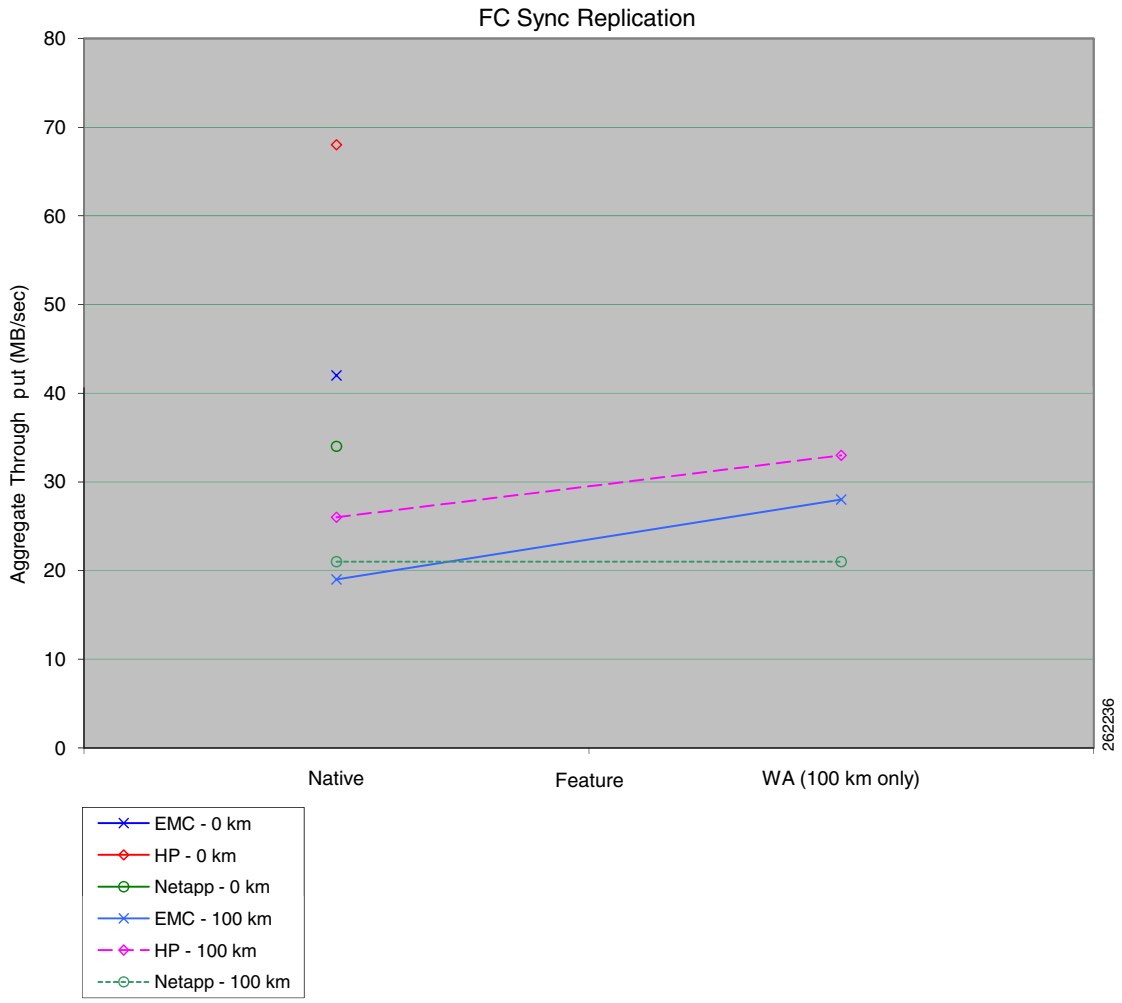


Figure 9-11 summarizes asynchronous FCIP replication results. Note that write acceleration doesn't benefit HP CA Journal because data transfer is done with reads rather than writes. Also note that Network Appliance SnapMirror doesn't benefit from write acceleration due to use of the IPFC and FC-VI protocols rather than FCP. Although EMC SRDF/A does benefit from write acceleration, replication sessions were suspended at the 5000 km distance for some or all hosts due to running out of storage frame cache, which skews the results.

Figure 9-11 Aynchronous FCIP Replication Result Summary

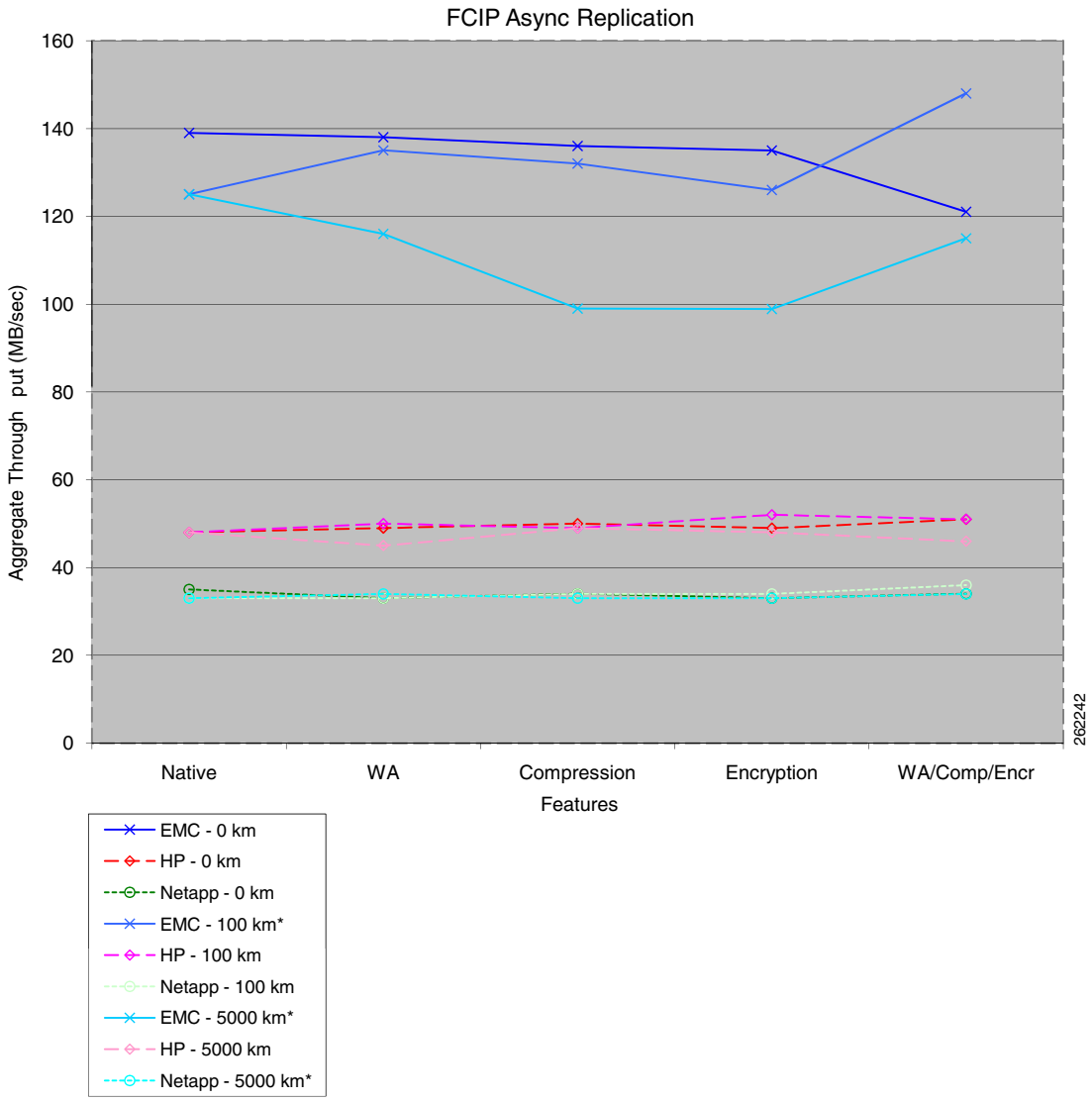


Figure 9-12 summarizes FCIP tape read acceleration results. As expected, tape acceleration provides the most benefit at the 5000 km distance. Hardware compression more than doubles throughput at all distances. Software compression halves throughput due to the overhead of software compression not being offset by bandwidth constraints (bandwidth is OC-3, 155 Mbps).

Figure 9-12 FCIP Tape Read Acceleration Result Summary

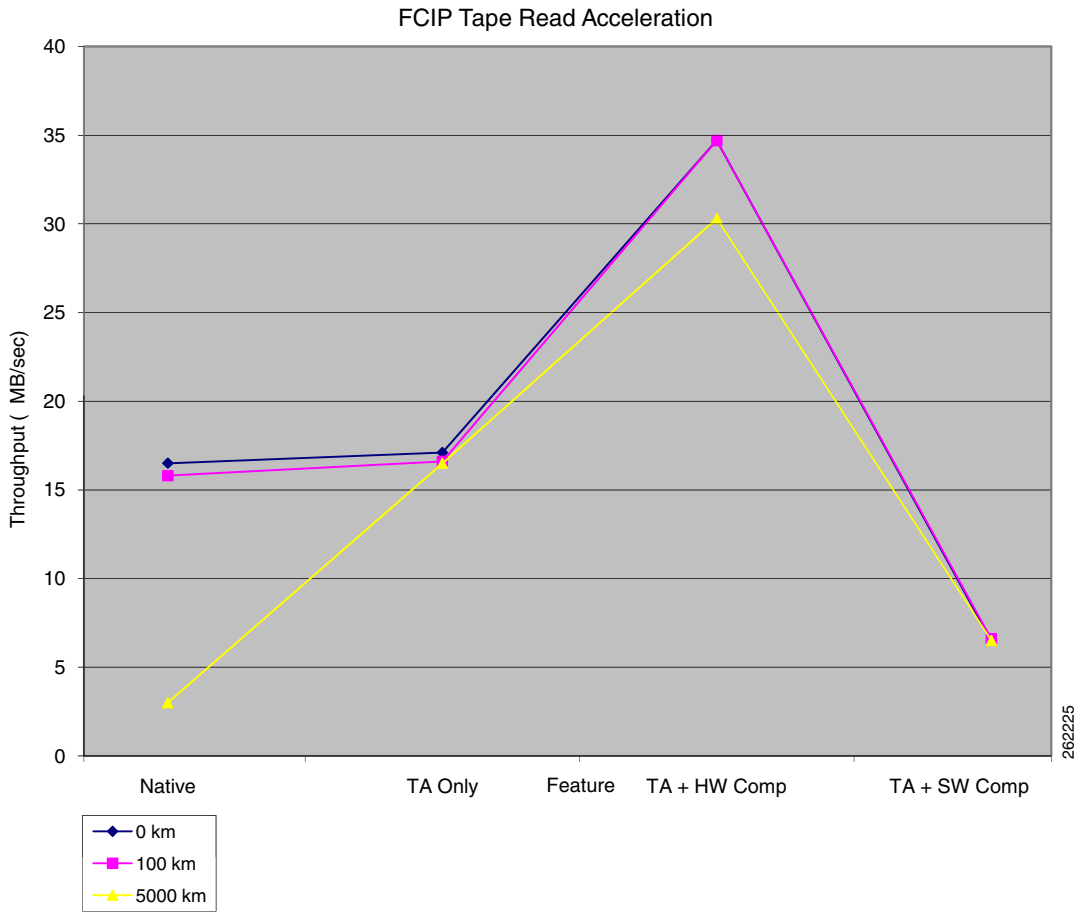
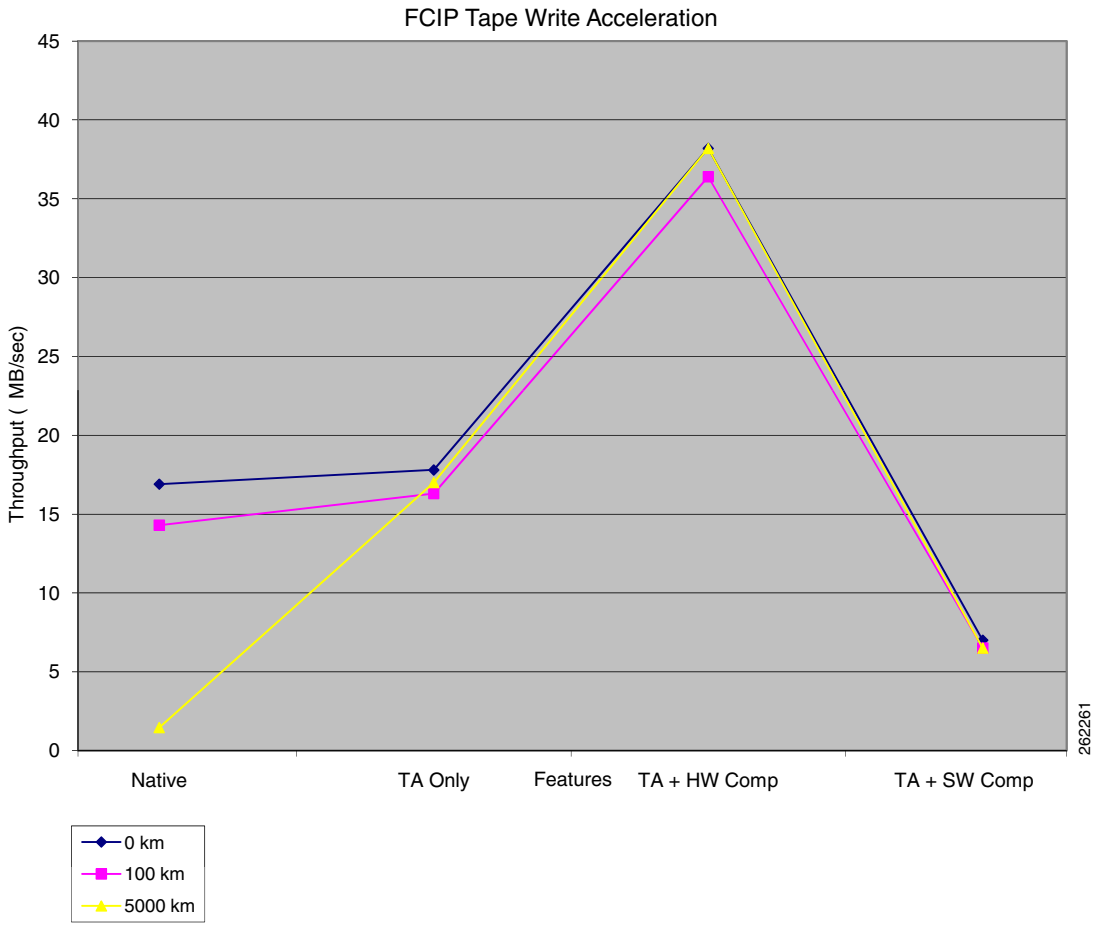


Figure 9-13 summarizes FCIP tape write acceleration results. As expected, tape acceleration provides the most benefit at the 5000 km distance. Hardware compression more than doubles throughput at all distances. Software compression halves throughput due to the overhead of software compression not being offset by bandwidth constraints (bandwidth is OC-3, 155 Mbps).

Figure 9-13 FCIP Tape Write Acceleration Result Summary



The Cisco DCAP 4.0 Storage Area Network (SAN) tests cover the MDS9500 platform and fall into four major categories: Baseline Testing, Functionality Testing, Resilience Testing, and SAN Extension Testing.

Baseline Testing looks at the general functionality and configuration of the devices in the DCAP SAN test topology. This includes ensuring each test host has redundant paths to each storage array and verifying replication paths between pairs of storage arrays are working properly. Configurations follow best practices for VSAN configuration, port allocation, zoning, and storage device mapping and masking. Functionality Testing covers key device management and Fiber Channel (FC) protocols such as virtual SAN (VSAN) configuration, port channels, Fabric Shortest Path First (FSPF), Inter-Virtual SAN routing (IVR), and security. Resilience Testing measures the response of the DCAP SAN topology to various failure conditions like cable pulls, power outage, component failures, and supervisor and module reloads and online insertions and replacements (OIRs). SAN Extension Testing focuses on replication of data between actual storage frames from EMC, Hewlett Packard, and Network Appliance using both the native FC protocol and the FC over Internet Protocol (FCIP) with simulated latencies. Advanced capabilities like FC write acceleration and FCIP compression, write acceleration, and encryption are included. SAN Extension Testing also includes remote backup and recovery of data to and from tape using the MDS FCIP tape read and write acceleration functionality with and without switch-based software and hardware compression.

The DCAP SAN topology is divided into three distinct, logical layers called the Transit, Core, and Edge layers offering services listed in Table 9-1.

Table 9-1 DCAP SAN Logical Layer Services

Logical Layer	Services
Transit	FC over SONET fabric extension (for synchronous storage replication), FCIP fabric extension (for asynchronous storage replication), DPVM, IVR (with and without NAT), FCIP compression and encryption, FC and FCIP write acceleration, FCIP tape acceleration
Core	Storage fabric connectivity
Edge	Host fabric connectivity
All Layers	VSANs, FSPF, port channels, zoning, port security, FC-SP, TACACS+ authentication

The DCAP SAN topology incorporates Cisco MDS fabric director products and design guides, industry best practices, and storage vendor implementation guidelines to provide a SAN infrastructure that is representative of the typical enterprise data center environment. The infrastructure provides both fundamental and advanced fiber channel SAN services for both host to storage connectivity and storage replication. The topology includes two separate data centers, each with dual, redundant core/edge fabrics for highly available and resilient connectivity, and dual transit fabrics that allow storage frame-based replication between the data centers for disaster recovery and business continuance. Fiber spools with 100 km optical fibers and Linux/netem-based IP delay generator allow simulation of distance between the data centers. For Phase 4, the tested distance is 100 km (62 miles), which corresponds to a round trip time of 1 ms, for FC, and both 100 km and 5000 km (80 ms round trip time) for FCIP. All fabrics use Cisco's Virtual SAN (VSAN) technology.

The transit fabric is the focal point of the topology and testing, since Cisco SAN extension capabilities are key differentiators for many customers. SAN extension is a key enabler for modern data center designs which call for redundant data centers. Cisco SAN extension supports both synchronous and asynchronous replication. The use of these terms follows industry practice; that is, synchronous replication means a host write transaction is not acknowledged by the primary or local storage frame until the secondary or remote storage frame confirms it has safely stored a replica of the data, and asynchronous replication means a successful host write is immediately acknowledged by the primary frame.

The FC transit core consists of of Cisco DS-CWDM-OADMA optical add/drop multiplexers which provide four bidirectional circuits over four separate wavelengths through a pair of 100 km optical fibers. Two circuits comprise the north transit fabric and the other two the south transit fabric. The circuits in each fabric support native FC traffic for synchronous replication. The IP transit core consists of connections to Catalyst 6500 access switches which are part of the data center LAN as well as WAN edge switches. All connectivity is with Gigabit Ethernet. The WAN switches are connected to each other through Linux hosts running network emulation (netem) software which allows latency generation for distance simulation and bandwidth limiting. The IP transit core supports FCIP traffic for asynchronous replication.

The primary SAN switch used in the topology is the Cisco MDS 9513 (model DS-C9513) with dual version 2 supervisors (model DS-X9530-SF2-K9) for maximum scaleability and nondisruptive operation and upgrades. Non-blade host connectivity is through high density 48-port FC modules (model DS-X9148) which provide 4 Gbps of bandwidth on an oversubscribed basis. Blade host connectivity for the HP cClass Blade Center is either through MDS 9124e (model DS-C9124-1-K9) switches or pass thru modules allowing connection to MDS 9513s. Storage connectivity is through lower density 12- and 24-port FC modules (models DS-X9112 and DS-X9124). Inter-switch links (ISLs) use 12-port FC modules to maximize bandwidth. All ISLs are in port channels. Connectivity between data centers relies on 32-port Storage Services Modules (model DS-X9032-SSM) for synchronous replication using FC

over CWDM and 14+2 port modules (model DS-X9302-14K9) for asynchronous replication using FC over IP (FCIP). The 14+2 modules have 14 FC interfaces capable of 2 Gbps and 2 Gigabit Ethernet interfaces.

The topology is designed to support testing that's as realistic as possible. Although test devices such as an Agilent SAN tester and Linux netem delay generators are part of the topology, these are only used when actual hosts and storage devices cannot provide suitable test conditions. To support this testing approach, each data center has FC storage frames from EMC (models DMX3 and CX3), Network Appliance (model FAS6070), and Hewlett Packard (model XP10000) as well as an FC tape library from Quantum (formerly ADIC, model Scalar i500). Each storage frame provides devices for primary application access as well as replication between data centers. The EMC DMX frames use Synchronous Replication Data Facility/Synchronous (SRDF/S) for synchronous replication and SRDF/Asynchronous (SRDF/A) for asynchronous replication. No replication testing was done on the EMC CX frames in this phase. The Network Appliance frames use SnapMirror for both types of replication. The Hewlett Packard frames use XP Continuous Access (CA) Synchronous for synchronous replication and and XP CA Journal for asynchronous replication. The backup and restore software used with the Quantum tape library is Veritas NetBackup on RedHat Enterprise Linux. More details for each vendor are in the appendix section.

Each data center also has both Windows 2003 and Linux RHEL 4 servers, both rack mount and blade, with multipath software and either 4-Gbps Qlogic or 2-Gbps or 4-Gbps Emulex HBAs providing two redundant paths to storage devices. Hosts accessing EMC storage use PowerPath for both Windows and Linux. Hosts with Network Appliance storage use ONTAP DSM for Windows and native MPIO (device-mapper-multipath) for Linux with the Network Appliance host attachment kit extras. Hosts with Hewlett Packard storage use HP MPIO DSM for Windows and native MPIO (device-mapper-multipath) for Linux. The majority of the storage tests are based on I/O generated by iometer on Windows and iorate on Linux. These tools are desirable because they're readily available in source code and binary distributions, easy to deploy and use, and provide basic performance information that helps determine the success or failure of tested MDS features. For Phase 4, both iometer and iorate were configured to do 8 KB sequential writes to generate load.

Test Results Summary

Table 9-2 on page 9-14 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 9-2 on page 9-14 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 9-2 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Baseline, page 9-20	Device Check, page 9-20	1. Device Access CLI and Device Manager 2. Device Hardware Check CLI 3. Device Hardware Check Device Manager 4. Device Network Services Check CLI 5. Device Network Services Check Device Manager	
Baseline, page 9-20	Host-To-Storage Traffic—EMC Clariion, page 9-25	1. Base Setup VSANs EMC CLARiiON 2. Base Setup Zoning EMC CLARiiON 3. Host To Storage IO Traffic EMC CLARiiON	CSCsk55538
Baseline, page 9-20	Host-To-Storage Traffic—EMC DMX, page 9-28	1. Base Setup VSANs EMC DMX 2. Base Setup Zoning EMC DMX 3. Host To Storage IO Traffic EMC DMX 4. Replication FC Sync EMC DMX 5. Replication FCIP Async EMC DMX	
Baseline, page 9-20	Host-To-Storage Traffic—HP XP, page 9-33	1. Base Setup VSANs HP XP 2. Base Setup Zoning HP XP 3. Host To Storage IO Traffic HP XP 4. Replication FC Sync HP XP 5. Replication FCIP Async HP XP 6. Replication FCIP Async Journal HP XP	
Baseline, page 9-20	Host-To-Storage Traffic—NetApp, page 9-40	1. Base Setup VSANs NetApp 2. Base Setup Zoning NetApp 3. Host To Storage IO Traffic NetApp 4. Replication FC Sync NetApp 5. Replication FCIP Async NetApp	

Table 9-2 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Baseline, page 9-20	Infrastructure Check, page 9-46	1. Host and Storage Fabric Connectivity EMC CLARiiON 2. Host and Storage Fabric Connectivity EMC DMX 3. Host and Storage Fabric Connectivity HP XP 4. Host and Storage Fabric Connectivity NetApp 5. Intra Fabric Connectivity 6. Topology Discovery Fabric Manager	
Domain Parameters, page 9-51	n/a	1. Principal Switch Selection	
Fabric Extension, page 9-52	Async Replication—EMC DMX, page 9-52	1. FCIP Compression EMC DMX 2. FCIP Encryption EMC DMX 3. FCIP Native EMC DMX 4. FCIP Port Channel Failure EMC DMX 5. FCIP Write Acceleration Compression Encryption EMC DMX 6. FCIP Write Acceleration EMC DMX	
Fabric Extension, page 9-52	Async Replication—HP XP, page 9-60	1. FCIP Compression HP XP 2. FCIP Encryption HP XP 3. FCIP Native HP XP 4. FCIP Port Channel Failure HP XP 5. FCIP Write Acceleration Compression Encryption HP XP 6. FCIP Write Acceleration HP XP	
Fabric Extension, page 9-52	Async Replication—NetApp, page 9-68	1. FCIP Compression NetApp 2. FCIP Encryption NetApp 3. FCIP Native NetApp 4. FCIP Port Channel Failure NetApp 5. FCIP Write Acceleration Compression Encryption NetApp 6. FCIP Write Acceleration NetApp	
Fabric Extension, page 9-52	Sync Replication—EMC DMX, page 9-76	1. FC Native EMC DMX 2. FC Write Acceleration EMC DMX 3. FC Write Acceleration Port Channel Failure EMC DMX	
Fabric Extension, page 9-52	Sync Replication—HP XP, page 9-79	1. FC Native HP XP 2. FC Write Acceleration HP XP 3. FC Write Acceleration Port Channel Failure HP XP	

Table 9-2 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Fabric Extension, page 9-52	Sync Replication—NetApp, page 9-83	1. FC Native NetApp 2. FC Write Acceleration NetApp 3. FC Write Acceleration Port Channel Failure NetApp	
FCIP Tape Acceleration, page 9-87	Tape Read Acceleration, page 9-87	1. FCIP Tape Read Acceleration 0 km Hardware Compression 2. FCIP Tape Read Acceleration 0 km No Compression 3. FCIP Tape Read Acceleration 0 km Software Compression 4. FCIP Tape Read Acceleration 100 km Baseline 5. FCIP Tape Read Acceleration 100 km Hardware Compression 6. FCIP Tape Read Acceleration 100 km No Compression 7. FCIP Tape Read Acceleration 100 km Software Compression 8. FCIP Tape Read Acceleration 5000 km Baseline 9. FCIP Tape Read Acceleration 5000 km Hardware Compression 10. FCIP Tape Read Acceleration 5000 km No Compression 11. FCIP Tape Read Acceleration 5000 km Software Compression 12. FCIP Tape Read Acceleration Local Baseline 13. FCIP Tape Read Acceleration Remote Baseline	

Table 9-2 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
FCIP Tape Acceleration, page 9-87	Tape Write Acceleration, page 9-100	1. FCIP Tape Write Acceleration 0 km Hardware Compression 2. FCIP Tape Write Acceleration 0 km No Compression 3. FCIP Tape Write Acceleration 0 km Software Compression 4. FCIP Tape Write Acceleration 100 km Baseline 5. FCIP Tape Write Acceleration 100 km Hardware Compression 6. FCIP Tape Write Acceleration 100 km No Compression 7. FCIP Tape Write Acceleration 100 km Software Compression 8. FCIP Tape Write Acceleration 5000 km Baseline 9. FCIP Tape Write Acceleration 5000 km Hardware Compression 10. FCIP Tape Write Acceleration 5000 km No Compression 11. FCIP Tape Write Acceleration 5000 km Software Compression 12. FCIP Tape Write Acceleration Local Baseline 13. FCIP Tape Write Acceleration Remote Baseline	
FSPF Functionality, page 9-113	n/a	1. Basic FSPF Load Balancing 2. Path Selection Cost change on Equal Cost Paths 3. Primary Path Failure 4. Primary Path Removal VSAN Remove	
Inter-VSAN Routing Functionality, page 9-117	n/a	1. Basic IVR Implementation 2. Basic IVR NAT Implementation	
Port-Channel Functionality, page 9-119	n/a	1. Basic Port-Channel Load Balancing 2. Multiple Link ADD to Group 3. Multiple Links Failure in Group 4. Multiple Links Remove from Group 5. Single Link ADD to Group 6. Single Link Failure in Group 7. Single Link Remove from Group	

Table 9-2 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Resiliency Functionality, page 9-125	n/a	1. ACTIVE Crossbar Fabric Failover (OIR) 2. ACTIVE Supervisor Failover (OIR) 3. ACTIVE Supervisor Failover (Reload) 4. ACTIVE Supervisor Failover (manual-CLI) 5. Back Fan Tray Failure (Removal) 6. Core Facing Module Failure (OIR) 7. Core Facing Module Failure (Reload) 8. Front FAN TRAY Failure (Removal) 9. Node Failure (Power Loss) 10. Node Failure (Reload) 11. Power Supply Failure (Cord Removal) 12. Power Supply Failure (PowerOff) 13. Power Supply Failure (Removal) 14. SAN OS Code Upgrade Event 15. EMC DMX 16. STANDBY Supervisor Failure (Reload) 17. Unused Module Failure (OIR)	CSCsk96269
Resiliency Functionality, page 9-125	EMC Clariion, page 9-142	1. Host Facing Module Failure (OIR) EMC CLARiiON 2. Host Facing Module Failure (Reload) EMC CLARiiON 3. Host Link Failure (Link pull)—EMC CLARiiON 4. Host Link Failure (Port Shutdown) EMC CLARiiON	
Resiliency Functionality, page 9-125	EMC DMX, page 9-145	1. Host Link Failure Link Pull EMC DMX 2. Host Link Failure Port Shutdown EMC DMX 3. Host Module Failure OIR EMC DMX 4. Host Module Failure Reload EMC DMX	
Resiliency Functionality, page 9-125	HP XP, page 9-150	1. Host Link Failure Link Pull HP XP 2. Host Link Failure Port Shutdown HP XP 3. Host Module Failure OIR HP XP 4. Host Module Failure Reload HP XP	
Resiliency Functionality, page 9-125	NetApp, page 9-154	1. Host Link Failure Link Pull NetApp 2. Host Link Failure Port Shutdown NetApp 3. Host Module Failure OIR NetApp 4. Host Module Failure Reload NetApp	

Table 9-2 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Security Functionality, page 9-158	n/a	1. FC SP Authentication Failure 2. Port Security Basic Implementation 3. User Access TACACS Basic Test 4. User Access TACACS Servers Failure	
Zone Scalability, page 9-161	n/a	1. Maximum Zone Members (Basic Zoning with Device Alias) 2. Maximum Zone Members (Basic Zoning with PWWN)	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Baseline

Baseline tests verify network is in working order prior to starting testing and quantify steady state network performance.

This section contains the following topics:

Device Check

This section contains the following topics:

Device Access CLI and Device Manager

Individual devices/nodes in the test bed require multiple access methods for management purposes. These access methods include CLI and fabric manager/device manager. This test validated the support and availability of access via the console, telnet, SSH, and the device manager application. Access methods were executed from a management station with IP access to the devices and terminal servers.

Test Procedure

The procedure used to perform the Device Access CLI and Device Manager test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that each node can be accessed and logged in to via the console.

Step 3 Verify that each node can be accessed and logged into via telnet.

Step 4 Verify that each node can be accessed and logged into via SSH.

Step 5 Verify that each node can be accessed and logged into via device manager. (Screendump shown for only one switch.)

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect access support for console, telnet, SSH, and device manager to be active and operational.
- We expect local authorization/authentication to be supported by the access methods under validation without problems or issues.
- We expect no CPU or memory problems.

Results

Device Access CLI and Device Manager passed.

Device Hardware Check CLI

All hardware components of each device must be in active and operational status prior to the start of any test activities. This test verified that linecards or modules, redundant components, power supply units, and other environmental conditions were without problems or issues in each device.

Test Procedure

The procedure used to perform the Device Hardware Check CLI test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Access each MDS node in each fabric and verify that all linecards and modules are in operational OK condition.

Step 3 Verify that all environmental-related hardware and conditions are in operational OK status.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all linecards or modules to be in OK operational condition.
- We expect all environmental-related hardware (for example, fans, power supply units) to be in OK condition and fully operational.

Results

Device Hardware Check CLI passed.

Device Hardware Check Device Manager

All hardware components in each device must be in active and operational status prior to the start of any test activities. Using the device manager application, this test verified that linecards or modules, redundant components, power supply units, and other environmental conditions were without problems or issues in each device prior to test.

Test Procedure

The procedure used to perform the Device Hardware Check Device Manager test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Use device manager to access each MDS node in each fabric and verify that all linecards and modules are in operational OK condition. (Screendump from only one switch shown.)

Step 3 Use device manager to check and verify that all power and environmental related hardware and status are in operational OK condition. (Screendump from only one switch shown.)

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect hardware and environmental status of the nodes to be reviewed and validated via device manager.
- We expect all linecards or modules to be in OK operational condition.

- We expect all environmental related hardware (for example, fans, power supply units) to be in OK condition and fully operational.

Results

Device Hardware Check Device Manager passed.

Device Network Services Check CLI

Devices or nodes in the fabrics are required to have a common clock source via NTP, SNMP services for remote management and traps, and logging capabilities to remote SYSLOG servers. This test case verified that network services (NTP, SNMP, SYSLOG) were configured and operational in all nodes.

Test Procedure

The procedure used to perform the Device Network Services Check CLI test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify NTP is configured properly (that is, peers are the correct NTP servers) and operational (that is, statistics are working) in all nodes.

Step 3 Verify timestamping is synchronized in all nodes.

Step 4 Verify SNMP (including traps) is configured properly (that is, SNMP server IP addresses are correct, traps are enabled, communities set) and operational.

Step 5 Verify IP connectivity to SNMP servers.

Step 6 Verify SYSLOG (logging server) is configured properly (that is, SYSLOG server IP addresses are correct, timestamps = milliseconds) and operational.

Step 7 Verify IP connectivity to SYSLOG servers.

Step 8 Verify SYSLOG functionality by getting into and then out of configuration mode. This will generate a syslog message. Check the log in the syslog server to validate its delivery.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect NTP services to be configured and operational in each node within the tesbed.
- We expect SNMP services to be configured and operational in each node within the tesbed.
- We expect SYSLOG services to be configured and operational in each node within the tesbed.

Results

Device Network Services Check CLI passed.

Device Network Services Check Device Manager

Devices or nodes in the fabrics are required to have a common clock source via NTP, SNMP services for remote management and traps, and logging capabilities to remote SYSLOG servers. Using the Device manager application this test case verified that network services (NTP, SNMP, SYSLOG) were configured and operational in all nodes.

Test Procedure

The procedure used to perform the Device Network Services Check Device Manager test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify NTP is configured properly (i.e., peers are the correct NTP servers) and operational (i.e., statistics are working) in all nodes.

Step 3 Verify time-stamping is synchronized in all nodes.

Step 4 Verify SNMP (and traps) is configured properly (i.e., SNMP server IP addresses are correct, traps are enabled, communities set) and operational.

Step 5 Verify trap-generation functionality by checking for recent fabric manager events. If there are not any, try generating an SNMP authentication failure. This will generate a trap. Check events in FM or DM.

Step 6 Verify SYSLOG (logging server) is configured properly (i.e., SYSLOG server IP addresses are correct, timestamps = milliseconds) and operational.

Step 7 Verify IP connectivity to SYSLOG servers.

Step 8 Verify SYSLOG functionality by getting into and then out of configuration mode. Check the log in the syslog server to validate its delivery.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the device manager can accurately verify the active operational state of the services in question.
- We expect the NTP services to be configured and operational in each node within the testbed.
- We expect SNMP services to be configured and operational in each node within the testbed.
- We expect SYSLOG services to be configured and operational in each node within the testbed.

Results

Device Network Services Check Device Manager passed.

Host-To-Storage Traffic—EMC Clariion

This section contains the following topics:

Base Setup VSANs EMC CLARiiON

Host-to-storage communication is the most essential and basic service that a SAN must provide. These services are made up of building blocks that include: VSAN port membership, zone membership, zoneset activation, and LUN masking. This test verified the basic configuration and activation of all VSAN's needed for host-to-storage between hosts (with multiple operating systems) and storage arrays. VSAN's were configured and verified via fabric manager (with CLI validation).

Test Procedure

The procedure used to perform the Base Setup VSANs EMC CLARiiON test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Two test hosts are selected each with a different operating system (Windows Enterprise 2003 Server and Linux RedHat Enterprise). Both hosts are dual homed to two separate fabrics (as per test topology—fabric A and B). [Pretest Condition Number 2] Storage arrays are dual homed to the host fabrics and to the replication fabrics. [Pretest Condition Number 3] Storage array's LUN masking should be configured to allow access from the test hosts to the proper (non-replicating) LUN's.

Step 3 Create one Windows host VSAN per fabric. Add the Windows host and corresponding storage arrays fabric ports to that VSAN as members.

Step 4 Check that Windows host and corresponding storage array fabric ports logged in again to the fabrics and into the FC Name Server under the correct Windows host VSAN.

Step 5 Create one Linux host VSAN per fabric. Add the Linux host and corresponding storage array fabric ports to that VSAN as members.

Step 6 Check that Linux host and matching storage array fabric ports logged in again to the fabrics and into the FC Name Server under the correct Linux host VSAN.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect fabric manager to configure all VSAN's between hosts and storage arrays without problems or issues.
- We expect no problems or issues with the configuration and verification of services' VSAN's.

- We expect all created VSAN's to be allowed and active in all port-channel/trunk ISL's/fabric extension links.
- We expect no CPU or memory problems.

Results

Base Setup VSANs EMC CLARiiON passed with exception. The following exceptions were noted: CSCsk55538.

Base Setup Zoning EMC CLARiiON

Host-to-storage communication is the most essential and basic service that a SAN must provide. These services are made up of building blocks that include: VSAN port membership, zone membership, zoneset activation, and LUN masking. This test verified the base zoning configuration to enable communication between hosts (with multiple operating systems) and storage arrays. Zones and zone sets were configured and verified via fabric manager (with CLI validation).

Test Procedure

The procedure used to perform the Base Setup Zoning EMC CLARiiON test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSAN configuration has been executed and validated in the Base Setup—VSAN's test case.

Step 3 For each fabric, create one Windows hosts zone for the Windows hosts VSAN. Add the Windows host and corresponding storage array fabric ports to that zone as members (that is, two member zone). Distribute the zone changes.

Step 4 Per fabric: Create one Linux hosts zone for the Linux hosts VSAN. Add the Linux host and matching storage arrays fabric ports to that zone as members (that is, two member zone). Distribute the zone changes.

Step 5 Per-fabric: Create a hosts zoneset and add the created zones. Activate and distribute the zoneset.

Step 6 Per-fabric: Verify zoneset distribution and activation across the fabric.

Step 7 Verify that each test host can see the required LUN's.

Step 8 Verify that each test-host's multipathing software can see the redundant paths available to it.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect fabric manager to configure all zones between hosts without problems or issues.
- We expect no problems or issues with the configuration and verification of services' zones.
- We expect all zone and zone members to be active and all zones distributed among nodes within the fabrics.

Results

Base Setup Zoning EMC CLARiiON passed.

Host To Storage IO Traffic EMC CLARiiON

Host-to-storage communication is based on input/output (IO) operations in which the host reads from and writes to the LUN's in the storage array. This test verified the communication (IO's) between hosts (with multiple operating systems) and a storage array. Traffic was generated with IOMETER (Windows) and IORATE (Linux). All test traffic ran over the VSANs and zones already configured and tested. The traffic statistics (IO Delay and IO per second) were observed, validated, and collected by CLI (with FM validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Iteration time : 5 minutes

Test Procedure

The procedure used to perform the Host To Storage IO Traffic EMC CLARiiON test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSAN's configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition Number 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case.

Step 3 Generate IO traffic from a test hosts (Windows and Linux) to the corresponding non replicated LUN's using the traffic characteristics defined in this test case.

Step 4 Verify using CLI that traffic is flowing without loss on the interfaces and load balanced.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport and delivery of all IO traffic between test hosts and storage arrays.
- We expect fabric manager and the CLI to be able to present accurate link utilization.
- We expect a logical distribution between read/write ratios and IOPS.

Results

Host To Storage IO Traffic EMC CLARiiON passed.

Host-To-Storage Traffic—EMC DMX

The host-to-storage traffic tests for EMC DMX ensure hosts can access storage devices andd that both SRDF/S for synchronous replication and SRDF/A for asynchronous replication are working properly.

This section contains the following topics:

Base Setup VSANs EMC DMX

Host-to-storage communication is the most essential and basic service that a SAN must provide, followed by replication (storage to storage for business continuance). These services are made up of building blocks that include: VSAN port membership, zone membership, zoneset activation, and LUN masking. This test verified the basic configuration and activation of all VSANs needed for host-to-storage and replication communication between hosts (with multiple operating systems), storage arrays, and storage array pair. VSANs were configured and verified via Fabric Manager (with CLI validation).

Test Procedure

The procedure used to perform the Base Setup VSANs EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Two test hosts are selected each with a different operating system (Windows Enterprise 2003 Server and Linux RedHat Enterprise). Both hosts are dual homed to two separate fabrics (as per test topology—fabric A and B, or fabric C and D). [Pretest Condition Number 2] Storage arrays are dual homed to the host fabrics and to the replication fabrics. [Pretest Condition Number 3] Storage array's LUN masking should be configured to allow access from the test hosts to the proper (non-replicating) LUN's. [Pretest Condition Number 4] Storage array's replication services must be enabled for sync and async replication between selected LUN's.

Step 3 Create one Windows host VSAN per fabric. Add the Windows host and corresponding storage arrays fabric ports to that VSAN as members.

Step 4 Check that Windows host and corresponding storage array fabric ports logged in again to the fabrics and into the FC Name Server under the correct Windows host VSAN.

Step 5 Create one Linux host VSAN per fabric. Add the Linux host and corresponding storage array fabric ports to that VSAN as members.

Step 6 Check that Linux host and matching storage array fabric ports logged in again to the fabrics and into the FC Name Server under the correct Linux host VSAN.

Step 7 Create two replication VSANs per transport fabric. Add the storage array's fabric ports to those VSANs as members.

Step 8 Check that the storage array and corresponding storage array replication ports logged in again to the transport fabrics and into the FC Name Server under the correct replication VSANs.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Fabric Manager to configure all VSANs between hosts and storage arrays and between storage arrays in the replication ports without problems or issues.
- We expect no problems or issues with the configuration and verification of services' VSANs.
- We expect all created VSANs to be allowed and active in all port-channel/trunk ISL's/fabric extension links.
- We expect no CPU or memory problems.

Results

Base Setup VSANs EMC DMX passed.

Base Setup Zoning EMC DMX

Host-to-storage communication is the most essential and basic service that a SAN must provide followed by replication (storage-to-storage for business continuance). These services are made up of building blocks that include: VSAN port membership, zone membership, zoneset activation, and LUN masking. This test verified the base zoning configuration to enable communication between hosts (with multiple operating systems) and storage arrays and between storage array pairs. Zones and zone sets were configured and verified via Fabric Manager (with CLI validation).

Test Procedure

The procedure used to perform the Base Setup Zoning EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSAN configuration has been executed and validated in the Base Setup—VSANs test case.

Step 3 For each fabric, create one Windows hosts zone for the Windows hosts VSAN. Add the Windows host and corresponding storage array fabric ports to that zone as members (that is, two member zone).

Step 4 Per fabric: Create one Linux hosts zone for the Linux hosts VSAN. Add the Linux host and matching storage arrays fabric ports to that zone as members (that is, two member zone).

Step 5 Per-replication fabric: Create one sync replication zone for the sync replication VSAN and one async replication zone for the async replication VSAN. Add the storage array ports to that zone as members (that is, two member zone).

Step 6 Per-fabric: Create a hosts zoneset and add the created zones. Activate and distribute the zoneset.

Step 7 Per-replication fabric: Create a replication zoneset and add the created zones. Activate and distribute the zoneset.

Step 8 Per-fabric: Verify zoneset distribution and activation across the fabric.

Step 9 Verify that each test host can see the required LUN's.

Step 10 Verify that each storage array can see the remote pair within the replication services.

Step 11 Verify that each test host's multipathing software can see the redundant paths available to it.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Fabric Manager to configure all zones between hosts and storage arrays and between storage arrays in the replication ports without problems or issues.
- We expect no problems or issues with the configuration and verification of services' zones.
- We expect all zone and zone members to be active and all zones distributed among nodes within the fabrics.

Results

Base Setup Zoning EMC DMX passed.

Host To Storage IO Traffic EMC DMX

Host-to-storage communication is based on input/output (IO) operations in which the host reads from and writes to the LUNs in the storage array. This test verified the communication (IO's) between hosts (with multiple operating systems) and a storage array. Traffic was generated with IOMETER (Windows) and IORATE (Linux). All test traffic ran over the VSANs and zones already configured and tested. The traffic statistics (IO Delay and IO per second) were observed, validated, and collected by CLI (with FM validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Iteration time : 5 minutes

Test Procedure

The procedure used to perform the Host To Storage IO Traffic EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition Number 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case.

Step 3 Generate IO traffic from test hosts (Windows and Linux) to the corresponding non-replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using CLI that traffic is flowing without loss on the interfaces and load balanced.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport and delivery of all IO traffic between test hosts and storage arrays.
- We expect Fabric Manager and the CLI to be able to present accurate link utilization.
- We expect a logical distribution between read/write ratios and IOPS.

Results

Host To Storage IO Traffic EMC DMX passed.

Replication FC Sync EMC DMX

Synchronous replication propagates host I/Os between an array local to a host and a remote array. The local array does not acknowledge a write to a host until the remote array acknowledges the replicated write. This test verified the basic functionality of synchronous replication between a storage array pair with I/O from both Linux and Windows hosts. The mechanism used is SRDF/S. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O Delay and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- No FCWA, encryption, compression
- Iteration time : 5 minutes
- Distance : 0 Km

Test Procedure

The procedure used to perform the Replication FC Sync EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition Number 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition Number 3] Host-to-storage testing was successfully executed.

Step 3 Generate IO traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using Fabric Manager and CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all IO traffic between test hosts and the storage array pair for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fall out of synchronization due to MDS related issues.
- We expect the IO delay statistics to be higher (that is, longer delay) and for less IOPS than the host-to-storage scenario.

Results

Replication FC Sync EMC DMX passed.

Replication FCIP Async EMC DMX

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is SRDF/A. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- No FCIP-WA, encryption, compression
- Iteration time : 5 minutes
- Distance : 0 Km

Test Procedure

The procedure used to perform the Replication FCIP Async EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be similar to the host-to-storage scenario.
- We expect no CPU or memory problems.

Results

Replication FCIP Async EMC DMX passed.

Host-To-Storage Traffic—HP XP

The host-to-storage traffic tests for HP XP ensure hosts can access storage devices andd that Continuous Access (CA) XP Sychronous for synchronous replication and CA XP Asynchronous and CA XP Journal for asynchronous replication are working properly.

This section contains the following topics:

Base Setup VSANs HP XP

Host-to-storage communication is the first most essential and basic service that a SAN must provide followed by replication (storage-to-storage for Business Continuance). These services are made up of building blocks which include: VSAN port membership, zone membership, zoneset activation, LUN masking, etc. This test verified the basic configuration and activation of all VSANs needed for host-to-storage and replication communication between hosts (with multiple operating systems), storage arrays, and storage array pair. VSANs were configured and verified via Fabric Manager (with CLI validation).

Test Procedure

The procedure used to perform the Base Setup VSANs HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pre-Test Condition #1] Two test hosts are selected each with a different operating system (Windows Enterprise 2003 Server and Linux RedHat Enterprise). Both hosts are dual-homed to two separate fabrics (as per test topology—Fabric A and B -OR- Fabric C and D). [Pre-Test Condition #2] Storage Arrays are dual-homed to the host fabrics and to the replication fabrics. [Pre-Test Condition #3] Storage array's LUN masking should be configured to allow access from the test hosts to the proper (non-replicating) LUNs. [Pre-Test Condition #4] Storage array's replication services must be enabled for sync and async replication between selected LUNs.

Step 3 Create one Windows host VSAN per fabric. Add the Windows host and corresponding storage arrays fabric ports to that VSAN as members.

Step 4 Check that Windows host and corresponding storage array fabric ports re-login into the fabrics and into the FC Name Server under the correct Windows host VSAN.

Step 5 Create one Linux host VSAN per fabric. Add the Linux host and corresponding storage array fabric ports to that VSAN as members.

Step 6 Check that Linux host and matching storage array fabric ports re-login into the fabrics and into the FC Name Server under the correct Linux host VSAN.

Step 7 Create two replication VSANs per transport fabric. Add the storage array's fabric ports to those VSANs as members.

Step 8 Check that the storage array and corresponding storage array replication ports re-login into the transport fabrics and into the FC Name Server under the correct replication VSANs.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that Fabric Manager is able to configure all VSANs between hosts and storage arrays and between storage arrays in the replication ports without problems or issues.
- We expect no problems or issues with the configuration and verification of services' VSANs.
- We expect all created VSANs to be allowed and active in all Port-Channel / trunk ISLs / Fabric Extension links.
- We expect no CPU or memory problems.

Results

Base Setup VSANs HP XP passed.

Base Setup Zoning HP XP

The host-to-storage communication is the first most essential and basic service that a SAN must provide followed by replication (storage-to-storage for business continuance). These services are made up of building blocks which include: VSAN port membership, zone membership, zoneset activation, LUN masking, etc. This test verified the base zoning configuration to enable communication between hosts (with multiple operating systems) and storage arrays and between storage array pairs. Zones and zone sets were configured and verified via Fabric Manager (with CLI validation).

Test Procedure

The procedure used to perform the Base Setup Zoning HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSAN configuration has been executed and validated in the Base Setup—VSANs test case.

Step 3 For each fabric, create one Windows hosts zone for the Windows hosts VSAN. Add the Windows host and corresponding storage array fabric ports to that zone as members (that is two member zone).

Step 4 Per fabric: Create one Linux hosts zone for the Linux hosts VSAN. Add the Linux host and matching storage arrays fabric ports to that zone as members (that is, two member zone).

Step 5 Per replication fabric: Create one sync replication zone for the sync replication VSAN and one async replication zone for the async replication VSAN. Add the storage array ports to that zone as members (that is, two member zone).

Step 6 Per fabric: Create a hosts zone set and add the created zones. Activate and distribute the zone set.

Step 7 Per replication fabric: Create a replication zone set and add the created Zones. activate and distribute the zone set.

Step 8 Per fabric: Verify zone set distribution and activation across the fabric.

Step 9 Verify that each test host can see the required LUNs.

Step 10 Verify that each storage array can see the remote pair within the replication services.

Step 11 Verify that each test host's multi pathing software can see the redundant paths available to it.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Fabric Manager to be able to configure all zones between hosts and storage arrays and between storage arrays in the replication ports without problems or issues.
- We expect no problems or issues with the configuration and verification of services' zones.
- We expect all zone and zone members to be active and all zones distributed among nodes within the fabrics.

Results

Base Setup Zoning HP XP passed.

Host To Storage IO Traffic HP XP

Host-to-storage communication is based on input/output (IO) operations in which the host reads from and writes to the LUNs in the storage array. This test verified the communication (IOs) between hosts (with multiple operating systems) and a storage array. Traffic was generated with IOMETER (Windows) and IORATE (Linux). All test traffic ran over the VSANs and zones already configured and tested. The traffic statistics (IO Delay and IO per second) were observed, validated, and collected by CLI (with FM validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Iteration time : 5 minutes

Test Procedure

The procedure used to perform the Host To Storage IO Traffic HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition Number 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case.

Step 3 Generate IO traffic from a test hosts (Windows and Linux) to the corresponding non-replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport and delivery of all IO traffic between test hosts and storage array.
- We expect for the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect a logical distribution between read/write ratios and IOPS.
- We expect no CPU or memory problems.

Results

Host To Storage IO Traffic HP XP passed.

Replication FC Sync HP XP

Synchronous replication propagates host I/Os between an array local to a host and a remote array. The local array does not acknowledge a write to a host until the remote array acknowledges the replicated write. This test verified the basic functionality of synchronous replication between a storage array pair with I/O from both Linux and Windows hosts. The mechanism used is Continuous Access XP Sync. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O Delay and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- No FCWA, encryption, compression
- Iteration time : 5 minutes
- Distance : 0 Km

Test Procedure

The procedure used to perform the Replication FC Sync HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition Number 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition Number 3] Host-to-storage testing was successfully executed.

Step 3 Generate IO traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using Fabric Manager and CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all IO traffic between test hosts and the storage array pair for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fall out of synchronization due to MDS related issues.
- We expect the IO delay statistics to be higher (that is, longer delay) and for less IOPS than the host-to-storage scenario.

Results

Replication FC Sync HP XP passed.

Replication FCIP Async HP XP

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is Continuous Access XP Async. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- No FCIP-WA, encryption, compression
- Iteration time : 5 minutes
- Distance : 0 Km

Test Procedure

The procedure used to perform the Replication FCIP Async HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition Number 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition Number 3] Host-to-storage testing was successfully executed. [Pretest Condition Number 4] Check for appropriate latency.

Step 3 Generate IO traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all IO traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.

- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the IO throughput statistics to be similar to the host-to-storage scenario.
- We expect no CPU or memory problems.

Results

Replication FCIP Async HP XP passed.

Replication FCIP Async Journal HP XP

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is Continuous Access XP Journal. Journal differs from Async in that data is "pulled" from the remote array (versus "pushed" from the local array with Async) and a journal volume replaces the side file used by Async. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (IO throughput and IO per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- No FCIP-WA, encryption, compression
- Iteration time : 5 minutes
- Distance : 0 Km

Test Procedure

The procedure used to perform the Replication FCIP Async Journal HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition Number 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition Number 3] Host-to-storage testing was successfully executed. [Pretest Condition Number 4] Check for appropriate latency.

Step 3 Generate IO traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all IO traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the IO throughput statistics to be similar to the host-to-storage scenario.
- We expect no CPU or memory problems.

Results

Replication FCIP Async Journal HP XP passed.

Host-To-Storage Traffic—NetApp

The host-to-storage traffic tests for HP XP ensure hosts can access storage devices andd that SnapMirror for synchronous and asynchronous replication are working properly.

This section contains the following topics:

Base Setup VSANs NetApp

Host-to-storage communication is the first most essential and basic service that a SAN must provide followed by replication (storage-to-storage for business continuance). These services are made up of building blocks which include: VSAN port membership, zone membership, zoneset activation, LUN masking, etc. This test verified the basic configuration and activation of all VSANs needed for host-to-storage and replication communication between hosts (with multiple operating systems), storage arrays, and storage array pair. VSAN's were configured and verified via Fabric Manager (with CLI validation).

Test Procedure

The procedure used to perform the Base Setup VSANs NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Two test hosts are selected each with a different operating system (Windows Enterprise 2003 Server and Linux RedHat Enterprise). Both hosts are dual homed to two separate fabrics (as per test topology—fabric A and B -OR- fabric C and D). [Pretest Condition Number 2] Storage Arrays are dual homed to the host fabrics and to the replication fabrics. [Pretest Condition

Number 3] Storage array's LUN masking should be configured to allow access from the test hosts to the proper (non-replicating) LUN's. [Pretest Condition Number 4] Storage array's replication services must be enabled for sync and async replication between selected LUN's.

Step 3 Create one Windows host VSAN per fabric. Add the Windows host and corresponding storage arrays fabric ports to that VSAN as members.

Step 4 Check that Windows host and corresponding storage array fabric ports relogin into the fabrics and into the FC name server under the correct Windows host VSAN.

Step 5 Create one Linux host VSAN per fabric. Add the Linux host and corresponding storage array fabric ports to that VSAN as members.

Step 6 Check that Linux host and matching storage array fabric ports relogin into the fabrics and into the FC name server under the correct Linux host VSAN.

Step 7 Create two replication VSANs per transport fabric. Add the storage array's fabric ports to those VSANs as members.

Step 8 Check that the storage array and corresponding storage array replication ports relogin into the transport fabrics and into the FC name server under the correct replication VSANs.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Fabric Manager to be able to configure all VSANs between hosts and storage arrays and between storage arrays in the replication ports without problems or issues.
- We expect no problems or issues with the configuration and verification of services' VSANs.
- We expect all created VSANs to be allowed and active in all port-channel/trunk ISL's/fabric extension links.
- We expect no CPU or memory problems.

Results

Base Setup VSANs NetApp passed.

Base Setup Zoning NetApp

The host-to-storage communication is the first most essential and basic service that a SAN must provide followed by replication (storage-to-storage for business continuance). These services are made up of building blocks which include: VSAN port membership, zone membership, zoneset activation, LUN masking, etc. This test verified the base zoning configuration to enable communication between hosts (with multiple operating systems) and storage arrays and between storage array pairs. Zones and zone sets were configured and verified via Fabric Manager (with CLI validation).

Test Procedure

The procedure used to perform the Base Setup Zoning NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSAN configuration has been executed and validated in the Base Setup—VSANs test case.

Step 3 For each fabric, create one Windows hosts zone for the Windows hosts VSAN. Add the Windows host and corresponding storage array fabric ports to that zone as members (that is two member zone).

Step 4 Per fabric: Create one Linux hosts zone for the Linux hosts VSAN. Add the Linux host and matching storage arrays fabric ports to that zone as members (that is, two member zone).

Step 5 Per replication fabric: Create one sync replication zone for the sync replication VSAN and one async replication zone for the async replication VSAN. Add the storage array ports to that zone as members (that is, two member zone).

Step 6 Per fabric: Create a hosts zone set and add the created zones. Activate and distribute the zone set.

Step 7 Per replication fabric: Create a replication zone set and add the created Zones. activate and distribute the zone set.

Step 8 Per fabric: Verify zone set distribution and activation across the fabric.

Step 9 Verify that each test host can see the required LUN's.

Step 10 Verify that each storage array can see the remote pair within the replication services.

Step 11 Verify that each test host's multi pathing software can see the redundant paths available to it. (NOTE: Since the filers are running in single image mode, one path by default will be active and one passive. This may change in future versions of ONTAP DSM.)

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Fabric Manager to be able to configure all zones between hosts and storage arrays and between storage arrays in the replication ports without problems or issues.
- We expect no problems or issues with the configuration and verification of services' zones.
- We expect all zone and zone members to be active and all zones distributed among nodes within the fabrics.

Results

Base Setup Zoning NetApp passed.

Host To Storage IO Traffic NetApp

Host-to-storage communication is based on input/output (IO) operations in which the host reads from and writes to the LUNs in the storage array. This test verified the communication (IO's) between hosts (with multiple operating systems) and a storage array. Traffic was generated with IOMETER (Windows) and IORATE (Linux). All test traffic ran over the VSANs and zones already configured and tested. The traffic statistics (IO Delay and IO per second) were observed, validated, and collected by CLI (with FM validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Iteration time : 5 minutes

Test Procedure

The procedure used to perform the Host To Storage IO Traffic NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition Number 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case.

Step 3 Generate IO traffic from test hosts (Windows and Linux) to the corresponding non replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport and delivery of all IO traffic between test hosts and storage array.
- We expect for the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect a logical distribution between read/write ratios and IOPS.
- We expect no CPU or memory problems.

Results

Host To Storage IO Traffic NetApp passed.

Replication FC Sync NetApp

Synchronous replication propagates host I/Os between an array local to a host and a remote array. The local array does not acknowledge a write to a host until the remote array acknowledges the replicated write. This test verified the basic functionality of synchronous replication between a storage array pair

with I/O from both Linux and Windows hosts. The mechanism used is synchronous snapmirror. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O Delay and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- No FCWA, encryption, compression
- Iteration time : 5 minutes
- Distance : 0 Km

Test Procedure

The procedure used to perform the Replication FC Sync NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition Number 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition Number 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync-replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using Fabric Manager and CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all IO traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fall out of sync due to MDS related issues.
- We expect the IO delay statistics to be higher (that is, longer delay) and less IOPS than the host-to-storage scenario.
- We expect no CPU or memory problems.

Results

Replication FC Sync NetApp passed.

Replication FCIP Async NetApp

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is asynchronous snapmirror. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics are as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- No FCIP-WA, encryption, compression
- Iteration time : 5 minutes
- Distance : 0 Km

Test Procedure

The procedure used to perform the Replication FCIP Async NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition Number 2] Base Zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition Number 3] Host-to-storage test successfully executed. [Pretest Condition Number 4] Check for appropriate latency.

Step 3 Generate IO traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all IO traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the IO throughput statistics to be similar to the host-to-storage scenario.
- We expect no CPU or memory problems.

Results

Replication FCIP Async NetApp passed.

Infrastructure Check

The infrastructure check tests ensure all hosts and storage devices are logged into the appropriate fabric, storage replication works, and Fabri Manager properly discovers all fabrics and reports no anomalies.

This section contains the following topics:

Host and Storage Fabric Connectivity EMC CLARiiON

The connectivity between test hosts, storage arrays, and the fabrics was tested to ensure a problem free infrastructure prior to testing. The verification was done by means of checking port status/conditions and complete fabric logins from the part of the end devices in all links available (for example, devices are dual homed in many instances). This was done via the CLI with fabric manager validation.

Test Procedure

The procedure used to perform the Host and Storage Fabric Connectivity EMC CLARiiON test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify all test host and storage array connections are active and displayed correctly in the topology map.

Step 3 Verify successful fabric logins, fcns registration, and device aliases by checking correct PWWN, HBAs' IDs, and aliases.

Step 4 Check all hosts and storage arrays fabric ports against errors.

Step 5 Validate FM information (previous steps) via the CLI.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect fabric manager's fabric discovery process to accurately present host and storage arrays' connectivity information.

- We expect all links between hosts and corresponding fabric nodes to be active (UP/UP). The same applies to the storage arrays links.
- We expect all test hosts and storage arrays to successfully log into the fabrics (flogi).

Results

Host and Storage Fabric Connectivity EMC CLARiiON passed.

Host and Storage Fabric Connectivity EMC DMX

The connectivity between test hosts, storage arrays, and the fabrics was tested to ensure a problem free infrastructure prior to testing. The verification was done by means of checking port status/conditions and complete fabric logins from the part of the end devices in all links available (for example, devices are dual homed in many instances). This was done via the CLI with Fabric Manager validation.

Test Procedure

The procedure used to perform the Host and Storage Fabric Connectivity EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify all test host and storage array connections are active and displayed correctly in the topology map.

Step 3 Verify successful fabric logins, FCNS registration, and device aliases by checking correct PWWN, HBAs IDs, and aliases.

Step 4 Check all hosts and storage arrays fabric ports against errors.

Step 5 Validate FM information (previous steps) via the CLI.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Fabric Manager's fabric discovery process to accurately present host and storage arrays' connectivity information.
- We expect all links between hosts and corresponding fabric nodes to be active (UP/UP). The same applies to the storage arrays links.
- We expect all test hosts and storage arrays to successfully log into the fabrics (flogi).

Results

Host and Storage Fabric Connectivity EMC DMX passed.

Host and Storage Fabric Connectivity HP XP

The connectivity between test hosts, storage arrays, and the Fabrics was tested to ensure a problem free infrastructure prior to testing. The verification was done by means of checking port status/conditions and complete fabric logins from the part of the end devices in all links available (e.g., devices are dual-homed in many instances). This was done via the Fabric Manager application (part of the discovery process) with CLI validation.

Test Procedure

The procedure used to perform the Host and Storage Fabric Connectivity HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that all test hosts' and storage arrays' connections are active and displayed correctly in the topology map.

Step 3 Verify successful fabric logins, FCNS registration, and Device Aliases by checking correct PWWN, HBAs' IDs, aliases.

Step 4 Check all hosts and storage arrays fabric ports for errors.

Step 5 Validate FM information (previous steps) via CLI.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that Fabric Manager's fabric discovery process accurately presents host and storage arrays' connectivity information.
- We expect that all links between hosts and corresponding fabric nodes are active (UP/UP). The same applies to the storage arrays links.
- We expect that all test hosts and storage arrays successfully log into the Fabrics. (flogi).
- We expect no CPU or memory problems.

Results

Host and Storage Fabric Connectivity HP XP passed.

Host and Storage Fabric Connectivity NetApp

The connectivity between test hosts, storage arrays, and the fabrics was tested to ensure a problem free infrastructure prior to testing. The verification was done by means of checking port status/conditions and complete fabric logins from the part of the end devices in all links available (that is, devices are dual homed in many instances). This was done via the Fabric Manager application (part of the discovery process) with CLI validation.

Test Procedure

The procedure used to perform the Host and Storage Fabric Connectivity NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that all test hosts' and storage arrays' connections are active and displayed correctly in the topology map.

Step 3 Verify successful fabric logins, FCNS registration, and Device Aliases by checking correct PWWN, HBA IDs, and aliases.

Step 4 Check all hosts and storage arrays fabric ports for errors.

Step 5 Validate FM information (previous steps) via the CLI.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that Fabric Manager's fabric discovery process accurately presents host and storage arrays' connectivity information.
- We expect that all links between hosts and corresponding fabric nodes are active (UP/UP). The same applies to the storage arrays links.
- We expect that all test hosts and storage arrays successfully log into the fabrics. (flogi).
- We expect no CPU or memory problems.

Results

Host and Storage Fabric Connectivity NetApp passed.

Intra Fabric Connectivity

Intra-fabric connections (for example, inter-switch links and port-channels) and operational conditions were tested to ensure proper end-to-end connectivity within stable fabrics. The accuracy or proper discovery/representation of such conditions via Fabric Manager was part of this test. The test and validation was executed using Fabric Manager and verified via the CLI.

Test Procedure

The procedure used to perform the Intra Fabric Connectivity test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that all links between fabric nodes are active and operational.

Step 3 Verify that all port-channels are active (for example, all intended members are active in the channel).

Step 4 Verify connectivity within the transport fabrics (north and south) over IP and over CWDM. (Screendump from only one VSAN shown.)

Step 5 Validate/confirm, via the CLI, all intra-fabric connectivity as verified with FM in the previous steps. Run the **show port-channel summary** command and verify total and operational port counts match.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Fabric Manager to successfully discover all intra-fabric links/connections with accurate status.
- We expect all links between fabric nodes to be active and operating without problems (that is, no errors).
- We expect all port-channels to be fully operational.
- We expect for all fabric nodes to be able to see each other.

Results

Intra Fabric Connectivity passed.

Topology Discovery Fabric Manager

This test verfied that the fabric manager and the device manager could accurately and completely discover all six fabrics and devices attached to them. The appropriate "Host and Storage Fabric Connectivity" test cases and the "Intrafabric Connectivity" test case must be executed before this test.

Test Procedure

The procedure used to perform the Topology Discovery Fabric Manager test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Start fabric manager and open each fabric.

Step 3 Verify that all MDS nodes, hosts, and storage arrays are discovered and accurately identified.

Step 4 Open device manager sessions to each node in each fabric and verify proper hardware layout and ID information. (Only one screendump shown.)

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect fabric manager to be able to discover all nodes in all six fabrics without problems or issues.
- We expect no CPU or memory problems.

Results

Topology Discovery Fabric Manager passed.

Domain Parameters

This section contains the following topics:

Principal Switch Selection

The configuration and verification of principal switch selection static parameters was tested. All configuration and verification was done via Fabric Manager with confirmation through the CLI.

Test Procedure

The procedure used to perform the Principal Switch Selection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition Number 1] Identify the core principal switch within the target test fabric.

Step 3 Configure a non-principal switch as the new principal switch (configure higher priority).

Step 4 Verify the principal switch configuration.

Step 5 Perform a domain restart to apply configuration.

Step 6 Verify the new principal switch is active as the principal switch. Check that the previous principal switch is subordinate.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect hard-coded configuration of principal switches across the fabric to be successful with no problems or issues.
- We expect detection, reporting, and validation to be successfully done with the Fabric Manager with confirmation from the CLI.
- We expect no CPU or memory problems.

Results

Principal Switch Selection passed.

Fabric Extension

The fabric extension tests check synchronous and asynchronous storage replication over the SAN topology. For each vendor, synchronous tests over FC and asynchronous tests over FCIP are performed with different capablities enabled on the transit fabric.

This section contains the following topics:

Async Replication—EMC DMX

The asynchronous replication test for EMC tests SRDF/A over FCIP without any advanced services, with just FCIP write acceleration, with just FCIP compression, with just FCIP encryption, and with all three advanced services at the same time.

This section contains the following topics:

FCIP Compression EMC DMX

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is SRDF/A. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF

- Compression : ON
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Compression EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify compression statistics to show the feature is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be higher than the non-compressed scenario and for bandwidth utilization to be less due to compression.
- We expect no CPU or memory problems.

Results

FCIP Compression EMC DMX passed.

FCIP Encryption EMC DMX

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between

hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is SRDF/A. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Encryption EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify IP encryption is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect encryption will not affect performance, i.e. the I/O throughput statistics to be similar to the unencrypted FCIP 100 km scenario.
- We expect no CPU or memory problems.

Results

FCIP Encryption EMC DMX passed.

FCIP Native EMC DMX

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is SRDF/A. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Native EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be less than both the host-to-storage and FCIP with no added latency scenarios.
- We expect no CPU or memory problems.

Results

FCIP Native EMC DMX passed.

FCIP Port Channel Failure EMC DMX

This test verified the resilience of the fabric extension network (over IP) when one port channel link failed while async replication was active. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : ON
- Compression : ON
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP Port Channel Failure EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without issues.

Step 7 Verify FCIP WA, compression, and encryption are operational.

Step 8 Fail one link of the FCIP port-channel towards the transport (IP fabric extension) fabric.

Step 9 Confirm that the port-channel link is down. Verify that the failure is detected and reported by the nodes to the management applications.

Step 10 Verify that replication traffic is traversing the remaining port-channel link.

Step 11 Re-establish the port-channel failed link.

Step 12 Verify failed link is reestablished as a member of the port-channel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the port-channel including the recovered link.

Step 14 Verify the storage arrays' asynchronous replication state throughout the failure and recovery.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues like the fabric extension port-channel failure.
- We expect the port-channel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FCIP Port Channel Failure EMC DMX passed.

FCIP Write Acceleration Compression Encryption EMC DMX

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is SRDF/A. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : ON
- Compression : ON
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Write Acceleration Compression Encryption EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCIP WA, compression, and encryption are operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be improved due to write acceleration (although this may be offset to some degree by compression and encryption overhead).
- We expect bandwidth utilization to improve with the use of the compression feature.
- We expect the IP encryption feature not to have an adverse effect on the I/O statistics.
- We expect the combination of write acceleration, compression, and encryption not to have an negative effect on FCIP's functionality or the storage traffic delivery.
- We expect no CPU or memory problems.

Results

FCIP Write Acceleration Compression Encryption EMC DMX passed.

FCIP Write Acceleration EMC DMX

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between

hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is SRDF/A. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : ON
- Compression : OFF
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Write Acceleration EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCIP WA statistics to show the feature is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be greater than the corresponding non-write accelerated test due to write acceleration.
- We expect no CPU or memory problems.

Results

FCIP Write Acceleration EMC DMX passed.

Async Replication—HP XP

The asynchronous replication test for HP tests HP Continuous Access XP Journal replication over FCIP without any advanced services, with just FCIP write acceleration, with just FCIP compression, with just FCIP encryption, and with all three advanced services at the same time.

This section contains the following topics:

FCIP Compression HP XP

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is Continuous Access XP Journal. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : ON
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Compression HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify compression statistics to show the feature is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be at least as great as the uncompressed scenario and for bandwidth utilization to be less due to compression.
- We expect no CPU or memory problems.

Results

FCIP Compression HP XP passed.

FCIP Encryption HP XP

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is Continuous Access XP Journal. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Encryption HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify IP encryption is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect encryption willl not affect performance, i.e. the I/O throughput statistics to be similar to the corresponding FCIP scenario without encryption.
- We expect no CPU or memory problems.

Results

FCIP Encryption HP XP passed.

FCIP Native HP XP

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is Continuous Access XP Journal. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100

- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 0,100,5000 Km

Test Procedure

The procedure used to perform the FCIP Native HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be less than both the host-to-storage and FCIP with no added latency scenarios.
- We expect no CPU or memory problems.

Results

FCIP Native HP XP passed.

FCIP Port Channel Failure HP XP

This test verified the resilience of the fabric extension network (over IP) when one port channel link fails while async replication is active. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : ON
- Compression : ON
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP Port Channel Failure HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without issues.

Step 7 Verify FCIP WA, compression, and encryption are operational.

Step 8 Fail one link off the FCIP port-channel towards the transport (IP fabric extension) fabric.

Step 9 Confirm that the port-channel link is down. Verify that the failure is detected and reported by the nodes to the management applications.

Step 10 Verify that replication traffic is traversing the remaining port-channel link.

Step 11 Reestablish the port-channel failed link.

Step 12 Verify failed link is reestablished as a member of the port-channel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the port-channel including the recovered link. NOTE: Since HP XP Continuous Access can only use SID/DID load balancing, it's normal for one FCIP link to be carrying most of the traffic.

Step 14 Verify the storage arrays' asynchronous replication state throughout the failure and recovery.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues like the fabric extension port-channel failure.
- We expect the port-channel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FCIP Port Channel Failure HP XP passed.

FCIP Write Acceleration Compression Encryption HP XP

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is SRDF/A. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : ON
- Compression : ON
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Write Acceleration Compression Encryption HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCIP WA, compression, and encryption are operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be improved due to write acceleration (although this may be offset to some degree by compression and encryption overhead).
- We expect bandwidth utilization to improve with the use of the compression feature.
- We expect the IP encryption feature not to have an adverse effect on the I/O statistics.
- We expect the combination of write acceleration, compression, and encryption not to have an negative effect on FCIP's functionality or the storage traffic delivery.
- We expect no CPU or memory problems.

Results

FCIP Write Acceleration Compression Encryption HP XP passed.

FCIP Write Acceleration HP XP

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is Continuous Access XP Journal. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K

- Write-Acceleration : ON
- Compression : OFF
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Write Acceleration HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCIP WA statistics to show the feature is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be about the same as the corresponding non-write accelerated test due to XP Journal issuing reads (pulling data from remote side) versus writes (pushing data from local side).
- We expect no CPU or memory problems.

Results

FCIP Write Acceleration HP XP passed.

Async Replication—NetApp

The asynchronous replication test for NetApp tests asynchronous SnapMirror over FCIP without any advanced services, with just FCIP write acceleration, with just FCIP compression, with just FCIP encryption, and with all three advanced services at the same time.

This section contains the following topics:

FCIP Compression NetApp

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is asynchronous snapmirror. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : ON
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Compression NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify compression statistics to show the feature is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be at least as great as the non-compressed scenario and for bandwidth utilization to be less due to compression.
- We expect no CPU or memory problems.

Results

FCIP Compression NetApp passed.

FCIP Encryption NetApp

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is asynchronous snapmirror. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Encryption NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths. NOTE: NetApp doesn't load balance across paths per LUN in the current test configuration.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify IP encryption is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect encryption willl not affect performance, i.e. the I/O throughput statistics to be similar to the corresponding non-encrypted scenario.
- We expect no CPU or memory problems.

Results

FCIP Encryption NetApp passed.

FCIP Native NetApp

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is asynchronous snapmirror. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Native NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths. NOTE: NetApp doesn't load balance across paths per LUN in the current test configuration.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be less than both the host-to-storage and FCIP with no added latency scenarios.
- We expect no CPU or memory problems.

Results

FCIP Native NetApp passed.

FCIP Port Channel Failure NetApp

This test verified the resilience of the fabric extension network (over IP) when one port channel link fails while async replication is active. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : ON
- Compression : ON
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP Port Channel Failure NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths. NOTE: NetApp doesn't load balance across paths per LUN in the current test configuration.

Step 6 Verify that the storage array pair is replicating the LUNs without issues.

Step 7 Verify FCIP-WA, compression, and encryption are operational.

Step 8 Fail one link off the FCIP port-channel towards the transport (IP fabric extension) fabric.

Step 9 Confirm that the port-channel link is down. Verify that the failure is detected and reported by the nodes to the management applications.

Step 10 Verify that replication traffic is traversing the remaining port-channel link.

Step 11 Reestablish the port-channel failed link.

Step 12 Verify failed link is reestablished as a member of the port-channel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the port-channel including the recovered link. NOTE: Since NetApp can only use SID/DID load balancing, it's normal for one FCIP link to be carrying most of the traffic.

Step 14 Verify the storage arrays' asynchronous replication state throughout the failure and recovery.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues like the fabric extension port-channel failure.
- We expect the port-channel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FCIP Port Channel Failure NetApp passed.

FCIP Write Acceleration Compression Encryption NetApp

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is SRDF/A. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : ON
- Compression : ON
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 0, 100, 5000 Km

Test Procedure

The procedure used to perform the FCIP Write Acceleration Compression Encryption NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths. NOTE: NetApp doesn't load balance across paths per LUN in the current test configuration.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCIP WA, compression, and encryption are operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to be improved due to write acceleration (although this may be offset to some degree by compression and encryption overhead).
- We expect bandwidth utilization to improve with the use of the compression feature.
- We expect the IP encryption feature not to have an adverse effect on the I/O statistics.
- We expect the combination of write acceleration, compression, and encryption not to have an negative effect on FCIP's functionality or the storage traffic delivery.
- We expect no CPU or memory problems.

Results

FCIP Write Acceleration Compression Encryption NetApp passed.

FCIP Write Acceleration NetApp

Asynchronous replication propagates host I/Os between an array local to a host and a remote array. The local array immediately acknowledges a write to a host without waiting for the remote array to acknowledge the replicated write. This test verified the basic functionality of async replication between hosts (with multiple operating systems) and a storage array pair. Traffic is generated with tools like IOMETER and IORATE. The replication mechanism used is asynchronous snapmirror. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/O per second) were observed, validated, and collected by the CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100

- Block Sizes per r/w ratio : 8K
- Write-Acceleration : ON
- Compression : OFF
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP Write Acceleration NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs' test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage testing was successfully executed. [Pretest Condition 4] Check for appropriate latency.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding async replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths. NOTE: NetApp doesn't load balance across paths per LUN in the current test configuration.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCIP WA statistics to show the feature is operational. NOTE: for NetApp, which uses FC-VI versus FCP, FCWA doesn't speed things up; in this test, just confirm it's not breaking anything.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size iterations within the read/write ratio.
- We expect the CLI and Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fail their replication procedures due to MDS related issues.
- We expect the I/O throughput statistics to about the same as the host-to-storage statistics due to NetApp's IPFC-based SnapMirror implementation doesn't allow write acceleration to have any effect.
- We expect no CPU or memory problems.

Results

FCIP Write Acceleration NetApp passed.

Sync Replication—EMC DMX

The synchronous replication test for EMC tests SRDF/S with and without FC write acceleration.

This section contains the following topics:

FC Native EMC DMX

Synchronous replication propagates host I/Os between an array local to a host and a remote array. The local array does not acknowledge a write to a host until the remote array acknowledges the replicated write. This test verified the basic functionality of synchronous replication between a storage array pair with I/O from both Linux and Windows hosts. The mechanism used is SRDF/S. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O Delay and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- FCWA : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Native EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using Fabric Manager and CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size and read/write ratios.
- We expect Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fall out-of-sync due to MDS related issues.
- We expect the I/O throughput and IOPS to be lower than both the host-to-storage and undelayed extension scenarios.
- We expect no CPU or memory problems.

Results

FC Native EMC DMX passed.

FC Write Acceleration EMC DMX

Synchronous replication propagates host I/Os between an array local to a host and a remote array. The local array does not acknowledge a write to a host until the remote array acknowledges the replicated write. Fiber channel write acceleration (FCWA) speeds up the process by proxying remote acknowledgements on the local replication switch. This test verified the basic functionality of synchronous replication using FCWA between a storage array pair with I/O from both Linux and Windows hosts. The mechanism used is SRDF/S. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- FCWA : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Write Acceleration EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using Fabric Manager and CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCWA feature is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size and read/write ratios.
- We expect Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fall out-of-sync due to MDS related issues.
- We expect the I/O throughput and IOPS to be higher than for the corresponding test without FCWA enabled.
- We expect no CPU or memory problems.

Results

FC Write Acceleration EMC DMX passed.

FC Write Acceleration Port Channel Failure EMC DMX

This test verified the resilience of the fabric extension network when one port-channel link fails while sync replication is active. The traffic statistics (I/O throughput and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- FCWA : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Write Acceleration Port Channel Failure EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual-paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCWA statistics to show the feature is operational.

Step 8 Fail one link of the FC port-channel within the Transport (fabric extension) Fabric.

Step 9 Confirm that the port-channel link is down. Verify that the failure is detected and reported by the nodes to the management applications.

Step 10 Verify that replication traffic is traversing the remaining port-channel link.

Step 11 Re-establish the port-channel failed link.

Step 12 Verify failed link is reestablished as a member of the port-channel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the port-channel including the recovered link.

Step 14 Verify that storage arrays remained in synch throughout the failure and recovery. No traffic loss during each iteration within the test.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size and read/write ratio combinations.
- We expect Fabric Manager to be able to present accurate link utilization.
- We expect for the storage array pair not to fall out-of-sync due to MDS related issues like the fabric extension port-channel failure.
- We expect for the port-channel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FC Write Acceleration Port Channel Failure EMC DMX passed.

Sync Replication—HP XP

The synchronous replication test for HP tests HP Continuous Access XP Synchronous replication with and without FC write acceleration.

This section contains the following topics:

FC Native HP XP

Synchronous replication propagates host I/Os between an array local to a host and a remote array. The local array does not acknowledge a write to a host until the remote array acknowledges the replicated write. This test verified the basic functionality of sync replication between hosts (with multiple operating systems) and a storage array pair. Traffic was generated with tools like IOMETER and IORATE. All test traffic ran over the VSANs and zones already configured and tested in previous tests. The traffic statistics (IO delay, IO per second) were observed, validated, and collected by Fabric Manager (with CLI validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- FCWA : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Native HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using Fabric Manager and CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size and read/write ratios.
- We expect Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fall out-of-sync due to MDS related issues.
- We expect the I/O throughput and IOPS to be lower than both the host-to-storage and undelayed extension scenarios.
- We expect no CPU or memory problems.

Results

FC Native HP XP passed.

FC Write Acceleration HP XP

Synchronous replication propagates host I/Os between an array local to a host and a remote array. The local array does not acknowledge a write to a host until the remote array acknowledges the replicated write. Fiber channel write acceleration (FCWA) speeds up the process by proxying remote acknowledgements on the local replication switch. This test verified the basic functionality of synchronous replication using FCWA between a storage array pair with I/O from both Linux and Windows hosts. The mechanism used is Continuous Access XP Sync. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O Delay and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- FCWA : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Write Acceleration HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using Fabric Manager and CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCWA feature is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size and read/write ratios.

- We expect Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fall out-of-sync due to MDS related issues.
- We expect the I/O throughput and IOPS to be higher than for the corresponding test without FCWA enabled.
- We expect no CPU or memory problems.

Results

FC Write Acceleration HP XP passed.

FC Write Acceleration Port Channel Failure HP XP

This test verified the resilience of the fabric extension network when one port-channel link fails while sync replication is active. The traffic statistics (I/O throughput and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- FCWA : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Write Acceleration Port Channel Failure HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCWA statistics to show the feature is operational.

Step 8 Fail one link off the FC port-channel within the transport (fabric extension) fabric.

Step 9 Confirm that the port-channel link is down. Verify that the failure is detected and reported by the nodes to the management applications.

Step 10 Verify that replication traffic is traversing the remaining port-channel link.

Step 11 Reestablish the port-channel failed link.

Step 12 Verify failed link is reestablished as a member of the port-channel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the port-channel including the recovered link.

Step 14 Verify that storage arrays remained in sync throughout the failure and recovery. No traffic loss during each iteration within the test.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size and read/write ratio combinations.
- We expect Fabric Manager to be able to present accurate link utilization the storage array pair not to fall out-of-sync due to MDS related issues like the fabric extension port-channel failure.
- We expect the port-channel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FC Write Acceleration Port Channel Failure HP XP passed.

Sync Replication—NetApp

The synchronous replication test for NetApp tests synchronous SnapMirror with and without FC write acceleration.

This section contains the following topics:

FC Native NetApp

Synchronous replication propagates host I/Os between an array local to a host and a remote array. The local array does not acknowledge a write to a host until the remote array acknowledges the replicated write. This test verified the basic functionality of synchronous replication between a storage array pair with I/O from both Linux and Windows hosts. The mechanism used is synchronous snapmirror. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O Delay and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- FCWA : OFF

- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Native NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using Fabric Manager and CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths. NOTE: NetApp doesn't load balance across paths in the current test configuration for a single LUN.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size and read/write ratios.
- We expect Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fall out-of-sync due to MDS related issues.
- We expect the I/O throughput and IOPS to be lower than both the host-to-storage and undelayed extension scenarios.
- We expect no CPU or memory problems.

Results

FC Native NetApp passed.

FC Write Acceleration NetApp

Synchronous replication propagates host I/Os between an array local to a host and a remote array. The local array does not acknowledge a write to a host until the remote array acknowledges the replicated write. Fiber channel write acceleration (FCWA) speeds up the process by proxying remote acknowledgements on the local replication switch. However, this only holds true for native fiber channel. NetApp SnapMirror uses the FC-VI protocol rather than FCP, so FCWA does not cause a speed up. This test verified the basic functionality of synchronous replication using FCWA between a storage

array pair with I/O from both Linux and Windows hosts. The mechanism used is synchronous SnapMirror. All test traffic uses the VSANs and zones already configured and tested in previous tests. The traffic statistics (I/O throughput and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- FCWA : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Write Acceleration NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSAN configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify using Fabric Manager and CLI that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCWA feature is operational.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size and read/write ratios.
- We expect Fabric Manager to be able to present accurate link utilization.
- We expect the storage array pair not to fall out-of-sync due to MDS related issues.
- We expect the I/O throughput and IOPS to be about the same as for the corresponding test without FCWA enabled due to the IPFC-based implementation of synchronous snapmirror over fiber channel.
- We expect no CPU or memory problems.

Results

FC Write Acceleration NetApp passed.

FC Write Acceleration Port Channel Failure NetApp

This test verified the resilience of the fabric extension network when one port-channel link fails while sync replication is active. The traffic statistics (I/O throughput and I/Os per second) were observed, validated, and collected by CLI (with Fabric Manager validation) and the test tools. Traffic generation characteristics were as follows:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- FCWA : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Write Acceleration Port Channel Failure NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 [Pretest Condition 1] Base VSANs configuration has been executed and validated in the Base Setup—VSANs test case. [Pretest Condition 2] Base zones configuration has been executed and validated in the Base Setup—Zoning test case. [Pretest Condition 3] Host-to-storage test successfully executed.

Step 3 Generate I/O traffic from both hosts (Windows and Linux) to the corresponding sync replicated LUNs using the traffic characteristics defined in this test case.

Step 4 Verify (using Fabric Manager and CLI) that traffic is flowing without loss.

Step 5 Verify that the hosts are making use of the dual paths.

Step 6 Verify that the storage array pair is replicating the LUNs without problems or issues.

Step 7 Verify FCWA statistics to show the feature is operational.

Step 8 Fail one link off the FC port-channel within the transport (fabric extension) fabric.

Step 9 Confirm that the port-channel link is down. Verify that the failure is detected and reported by the nodes to the management applications.

Step 10 Verify that replication traffic is traversing the remaining port-channel link.

Step 11 Reestablish the port-channel failed link.

Step 12 Verify failed link is reestablished as a member of the port-channel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the port-channel including the recovered link.

Step 14 Verify that storage arrays remained in sync throughout the failure and recovery. No traffic loss during each iteration within the test.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect complete transport, delivery, and replication of all I/O traffic between test hosts and storage array pairs for all block size and read/write ratio combinations.
- We expect Fabric Manager to be able to present accurate link utilization the storage array pair not to fall out-of-sync due to MDS related issues like the fabric extension port-channel failure.
- We expect the port-channel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FC Write Acceleration Port Channel Failure NetApp passed.

FCIP Tape Acceleration

The FCIP tape acceleration tests check both read and write acceleration over varying simulated distances using an Quantum i500 Scalar tape library with IBM LTO3 drives and RedHat Enterprise Linux servers running Veritas NetBackup. Tests also include software and hardware compression.

This section contains the following topics:

Tape Read Acceleration

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. Testing involves restoring Windows and RedHat Linux operating system data and Microsoft Exchange and Oracle data over an FCIP link with tape acceleration enabled.

This section contains the following topics:

FCIP Tape Read Acceleration 0 km Hardware Compression

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test ensures the tape read acceleration feature is working over the shortest possibly distance by measuring the time for doing a restore from a tape drive to a host on a remote fabric connected by FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with no added latency and with tape acceleration and hardware compression enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 0 km Hardware Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with no added latency and fcip tape acceleration and hardware acceleration (mode 1) enabled (with all other advanced FCIP features disabled). Also ensure a restore image of the test data is available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access). Verify the tape acceleration feature was operational and compare the local and remote baseline throughput and time with this test's throughput and time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and the tape read acceleration function will work.
- We expect the throughput will be at least as great as the remote baseline test throughput and equal to or greater than the corresponding read acceleration test without compression.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 0 km Hardware Compression passed.

FCIP Tape Read Acceleration 0 km No Compression

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test ensures the tape read acceleration feature is working over the shortest possible distance by measuring the time for doing a restore from tape drives to a host on a remote fabric connected by FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with no added latency and tape acceleration enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 0 km No Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with no added latency and FCIP tape acceleration enabled (with all other advanced FCIP features disabled; note that enabling tape acceleration through FM also enables write acceleration). Also ensure a restore image of the test data is available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access). Verify the tape acceleration feature was operational and compare the local and remote baseline throughput and time with this test's throughput and time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and the tape read acceleration function will work.
- We expect the throughput will be at least as great as the remote baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 0 km No Compression passed.

FCIP Tape Read Acceleration 0 km Software Compression

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test ensures the tape read acceleration feature is working over the shortest possible distance by measuring the time for doing a restore from a tape drive to a host on a remote fabric connected by FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with no added latency and tape acceleration and software (mode 2) compression enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 0 km Software Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with no added latency and fcip tape acceleration and software acceleration (mode 2) enabled (with all other advanced FCIP features disabled). Also ensure a restore image of the test data is available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access). Verify the tape acceleration feature was operational and compare the local and remote baseline throughput and time with this test's throughput and time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and the tape read acceleration function will work.
- We expect the throughput will be at least as great as the remote baseline test throughput and equal to or greater than the corresponding read acceleration test without compression.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 0 km Software Compression passed.

FCIP Tape Read Acceleration 100 km Baseline

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test establishes baseline performance over a simulated 100 km distance by measuring the time for doing a restore from tape drives to a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 3 ms of added round trip latency (0.5 msec propagation delay and 1.0 msec device delay each way) without tape acceleration enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 100 km Baseline test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 3 ms of added latency and with FCIP tape acceleration and all other advanced FCIP features disabled. Also ensure a restore image of the test data is available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and take longer than the corresponding test cases with tape read acceleration enabled.
- We expect the throughput will be greater than the remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 100 km Baseline passed.

FCIP Tape Read Acceleration 100 km Hardware Compression

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test ensures the tape read acceleration feature is working over a simulated 100 km distance by measuring the time for doing a restore from tape drives to a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 3 ms of added round trip latency (0.5 msec propagation delay and 1.0 msec device delay each way) and tape acceleration and hardware (mode 1) compression enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 100 km Hardware Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 3 ms of added latency and fcip tape acceleration and hardware compression (mode 1) enabled (with all other advanced FCIP features disabled). Also ensure a restore image of the test data is available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access). Verify the tape acceleration feature was operational and compare the local and remote baseline throughput and time with this test's throughput and time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and the tape read acceleration function will work.
- We expect the throughput will be greater than the remote baseline test throughput and the corresponding tape acceleration with no compression test, but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 100 km Hardware Compression passed.

FCIP Tape Read Acceleration 100 km No Compression

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test ensures the tape read acceleration feature is working over a simulated 100 km distance by measuring the time for doing a restore from tape drives to a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 3 ms of added round trip latency (0.5 msec propagation delay and 1.0 msec device delay each way) and tape acceleration enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 100 km No Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 3 ms of added latency and fcip tape acceleration enabled (with all other advanced FCIP features disabled). Also ensure a restore image of the test data is available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access). Verify the tape acceleration feature was operational and compare the local and remote baseline throughput and time with this test's throughput and time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and the tape read acceleration function will work.
- We expect the throughput will be greater than the remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 100 km No Compression passed.

FCIP Tape Read Acceleration 100 km Software Compression

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test ensures the tape read acceleration feature is working over a simulated 100 km distance by measuring the time for doing a restore from tape drives to a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 3 ms of added round trip latency (0.5 msec propagation delay and 1.0 msec device delay each way) and tape acceleration and software (mode 2) compression enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 100 km Software Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drives on different fabrics with 3 ms of added latency and fcip tape acceleration and software compression (mode 2) enabled (with all other advanced FCIP features disabled). Also ensure a restore image of the test data is available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access). Verify the tape acceleration feature was operational and compare the local and remote baseline throughput and time with this test's throughput and time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and the tape read acceleration function will work.
- We expect the throughput will be greater than the remote baseline test throughput and the corresponding tape acceleration with no compression test, but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 100 km Software Compression passed.

FCIP Tape Read Acceleration 5000 km Baseline

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test establishes the baseline performance over a simulated 5000 km distance by measuring the time for doing a restore from tape drives to a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 80 ms of RTT latency (25 ms for propagation latency and 15 ms for device delay each way) without tape acceleration enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 5000 km Baseline test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drives on different fabrics with 80 ms of added latency and without FCIP tape acceleration or any other advance feature enabled. Also ensure a restore image of the test data is available and space is available on the host to restore the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed but that it will take much longer than when tape acceleration is enabled.
- We expect the throughput will be less than for the corresponding tests with tape acceleration enabled.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 5000 km Baseline passed.

FCIP Tape Read Acceleration 5000 km Hardware Compression

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test ensures the tape read acceleration feature is working over a simulated 5000 km distance by measuring the time for doing a restore from tape drives to a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 80 ms of RTT latency (25 ms for propagation latency and 15 ms for device delay each way) and tape acceleration and hardware compression enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 5000 km Hardware Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 80 ms of added latency and fcip tape acceleration and hardware compression (mode 1) enabled (with all other advanced FCIP features disabled). Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access). Verify the tape acceleration feature was operational and compare the local and remote baseline throughput and time with this test's throughput and time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and the tape read acceleration function will work.
- We expect the throughput will be greater than the corresponding remote baseline test throughput and the corresponding tape acceleration with no compression test, but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 5000 km Hardware Compression passed.

FCIP Tape Read Acceleration 5000 km No Compression

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test ensures the tape read acceleration feature is working over a simulated 5000 km distance by measuring the time for doing a restore from tape drives to a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 80 ms of RTT latency (25 ms for propagation latency and 15 ms for device delay each way) and tape acceleration enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 5000 km No Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drives on different fabrics with 80 ms of added latency and FCIP tape acceleration enabled (with all other advanced FCIP features disabled). Also ensure a restore image of the test data is available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access). Verify the tape acceleration feature was operational and compare the local and remote baseline throughput and time with this test's throughput and time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and the tape read acceleration function will work.
- We expect the throughput will be greater than the corresponding remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 5000 km No Compression passed.

FCIP Tape Read Acceleration 5000 km Software Compression

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test ensures the tape read acceleration feature is working over a simulated 5000 km distance by measuring the time for doing a restore from tape drives to a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 80 ms of RTT latency (25 ms for propagation latency and 15 ms for device delay each way) and tape acceleration and software compression (mode 2) enabled.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration 5000 km Software Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drives on different fabrics with 80 ms of added latency and FCIP tape acceleration and software compression (mode 2) enabled (with all other advanced FCIP features disabled). Also ensure restore images of the test data are available and space is available on the host to restore the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access). Verify the tape acceleration feature was operational and compare the local and remote baseline throughput and time with this test's throughput and time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and the tape read acceleration function will work.
- We expect the throughput will be greater than the correspnding remote baseline test throughput and the corresponding tape acceleration with no compression test, but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration 5000 km Software Compression passed.

FCIP Tape Read Acceleration Local Baseline

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test establishes a baseline time for the best possible performance by doing a restore from tapes to a host on the same fabric connected by a 4 Gbps fiber channel (not FCIP). The time taken for this restore will be the benchmark for later tests using FCIP with and without tape acceleration. This test also helps ensure the tape hardware and backup software are operating correctly.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration Local Baseline test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FC link between a host and 2 tape drives on the same fabric. Also ensure restore images of the test data are available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time. Use the "read time" in the job status (not the restore time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and establish a baseline for the maximum throughput and minimum time taken by a restore (tape read) of the test data.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration Local Baseline passed.

FCIP Tape Read Acceleration Remote Baseline

The FCIP tape acceleration feature speeds up restores (tape reads) by buffering reads at both ends of a possibly very long connection and allowing more than the usual single outstanding read. This test establishes a baseline time for doing a restore from tape drives to a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with no added latency and no

tape acceleration or any other advanced FCIP features enabled. The time taken for this restore will be the benchmark for later tests using FCIP with tape acceleration. This test also helps ensure the tape hardware and backup software are operating correctly.

Test Procedure

The procedure used to perform the FCIP Tape Read Acceleration Remote Baseline test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape restore setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with no added latency and no FCIP tape acceleration enabled. Also ensure a restore image of the test data is available and space is available on the host to restore the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a restore and monitor it for successful completion.

Step 5 After the restore is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were restored in what period of time.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the restore will succeed and establish a baseline for the maximum throughput and minimum time taken by a restore (tape read) of the test data over an FCIP link without any added latency and without tape acceleration enabled.
- We expect the throughput will be less than for the local baseline test.
- We expect no CPU or memory problems.

Results

FCIP Tape Read Acceleration Remote Baseline passed.

Tape Write Acceleration

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. Testing involves backing up Windows and RedHat Linux operating system data and Microsoft Exchange and Oracle data over an FCIP link with tape acceleration enabled.

This section contains the following topics:

FCIP Tape Write Acceleration 0 km Hardware Compression

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test ensures the tape write acceleration feature is working over the shortest possible distance by measuring the time for doing a backup to tape drives from a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with no added latency and tape acceleration and hardware compression enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 0 km Hardware Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with no added latency and FCIP tape acceleration and hardware compression (mode 1) enabled. Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and the tape write acceleration function will work.
- We expect the throughput will be close to the corresponding remote baseline test throughput but still less than the local baseline test.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 0 km Hardware Compression passed.

FCIP Tape Write Acceleration 0 km No Compression

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test ensures the tape write acceleration feature is working over the shortest possible distance by measuring the time for doing a backup to tape drives from a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with no added latency and tape acceleration enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 0 km No Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with no added latency and fcip tape acceleration enabled. Also ensure a filesystem containing the test data is available and tape media and a drive in the tape library are available to backup the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and the tape write acceleration function will work.
- We expect the throughput will be at least as great as the remote baseline test throughput.

- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 0 km No Compression passed.

FCIP Tape Write Acceleration 0 km Software Compression

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test ensures the tape write acceleration feature is working over the shortest possible distance by measuring the time for doing a backup to tape drives from a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with no added latency and tape acceleration and software compression enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 0 km Software Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with no added latency and FCIP tape acceleration and software compression (mode 2) enabled (with all other advanced FCIP features disabled). Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and the tape write acceleration function will work.
- We expect the throughput will close to the corresponding remote baseline test throughput but still less than the local baseline test.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 0 km Software Compression passed.

FCIP Tape Write Acceleration 100 km Baseline

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test establishes baseline performance over a simulated 100 km distance by measuring the time for doing a backup to tape drives from a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 3 ms of RTT latency (0.5 ms for propagation latency and 1 ms for device delay each way) and tape acceleration not enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 100 km Baseline test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 3 ms of added latency and FCIP tape acceleration and all other advanced FCIP features disabled. Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and take longer than the corresponding tests in which the tape write acceleration function is enabled.
- We expect the throughput will be less than when the tape write acceleration function is enabled.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 100 km Baseline passed.

FCIP Tape Write Acceleration 100 km Hardware Compression

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test ensures the tape write acceleration feature is working over a simulated 100 km distance by measuring the time for doing a backup to a tape drive from a host on a remote fabric connected by an FCIP using half of an OC-12 (max-bandwidth parameter is 311 Mbps) with 3 ms of RTT latency (0.5 ms for propagation latency and 1 ms for device delay each way) and tape acceleration and hardware compression enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 100 km Hardware Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drives on different fabrics with 3 ms of added latency and FCIP tape acceleration and hardware compression (mode 1) enabled (with all other advanced FCIP features disabled). Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and the tape write acceleration function will work.
- We expect the throughput will be near the remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 100 km Hardware Compression passed.

FCIP Tape Write Acceleration 100 km No Compression

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test ensures the tape write acceleration feature is working over a simulated 100 km distance by measuring the time for doing a backup to tape drives from a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 3 ms of RTT latency (0.5 ms for propagation latency and 1 ms for device delay each way) and tape acceleration enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 100 km No Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 3 ms of added latency and FCIP tape acceleration enabled (with all other advanced FCIP features disabled). Also ensure a filesystem containing the test data is available and tape media and a drive in the tape library are available to backup the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and the tape write acceleration function will work.
- We expect the throughput will be at least as great as the remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 100 km No Compression passed.

FCIP Tape Write Acceleration 100 km Software Compression

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test ensures the tape write acceleration feature is working over a simulated 100 km distance by measuring the time for doing a backup to a tape drive from a host on a remote fabric connected by an FCIP using half of an OC-12 (max-bandwidth parameter is 311 Mbps) with 3 ms of RTT latency (0.5 ms for propagation latency and 1 ms for device delay each way) and tape acceleration and software compression (mode 2) enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 100 km Software Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 3 ms of added latency and fcip tape acceleration and software compression (mode 2) enabled (with all other advanced FCIP features disabled). Also ensure a filesystem containing the test data is available and tape media and a drive in the tape library are available to backup the image. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and the tape write acceleration function will work.
- We expect the throughput will be at least as great as the remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 100 km Software Compression passed.

FCIP Tape Write Acceleration 5000 km Baseline

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test establishes baseline performance over a simulated 5000 km distance by measuring the time for doing a backup to tape drives from a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 80 ms of RTT latency (25 ms for propagation latency and 15 ms for device delay each way) and tape acceleration disabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 5000 km Baseline test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 80 ms of added latency and FCIPp tape acceleration and all other advanced FCIP features disabled. Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and will take longer than the corresponding test cases in which the tape write acceleration function is enabled.
- We expect the throughput will be less than the corresponding test cases in which the tape write acceleration function is enabled.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 5000 km Baseline passed.

FCIP Tape Write Acceleration 5000 km Hardware Compression

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test ensures the tape write acceleration feature is working over a simulated 5000 km distance by measuring the time for doing a backup to tape drives from a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 80 ms of RTT latency (25 ms for propagation latency and 15 ms for device delay each way) and tape acceleration and hardware compression enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 5000 km Hardware Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drives on different fabrics with 80 ms of added latency and FCIP tape acceleration and hardware compression (mode 1) enabled (with all other advanced FCIP features disabled). Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and the tape write acceleration function will work.
- We expect the throughput will be close to the corresponding remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 5000 km Hardware Compression passed.

FCIP Tape Write Acceleration 5000 km No Compression

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test ensures the tape write acceleration feature is working over a simulated 5000 km distance by measuring the time for doing a backup to tape drives from a host on a remote fabric connected by an FCIP using OC-3 bandwidth (max-bandwidth parameter is 155 Mbps) with 80 ms of RTT latency (25 ms for propagation latency and 15 ms for device delay each way) and tape acceleration enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 5000 km No Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 80 ms of added latency and fcip tape acceleration enabled (with all other advanced FCIP features disabled). Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and the tape write acceleration function will work.
- We expect the throughput will be nearly as great as the corresponding remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 5000 km No Compression passed.

FCIP Tape Write Acceleration 5000 km Software Compression

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test ensures the tape write acceleration feature is working over a simulated 5000 km distance by measuring the time for doing a backup to tape drives from a host on a remote fabric connected by an FCIP using OC-3 bsndwidth (max-bandwidth parameter is 155 Mbps) with 80 ms of RTT latency (25 ms for propagation latency and 15 ms for device delay each way) and tape acceleration and software compression (mode 2) enabled.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration 5000 km Software Compression test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drive on different fabrics with 80 ms of added latency and FCIP tape acceleration and software compression (mode 2) enabled (with all other advanced FCIP features disabled). Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server. Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and the tape write acceleration function will work.
- We expect the throughput will close to the corresponding remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration 5000 km Software Compression passed.

FCIP Tape Write Acceleration Local Baseline

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test establishes a baseline time for the best possible performance by doing a backup to tape drives from a host on the same fabric connected by a 4 Gbps fiber channel (not FCIP). The time taken for this backup will be the benchmark for later tests using FCIP with and without tape acceleration. This test also helps ensure the tape hardware and backup software are operating correctly.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration Local Baseline test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FC link between a host and 2 tape drives on the same fabric. Also ensure 2 filesystems containing the test data are available and tape media and 2 drives in the tape library are available to backup the data. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and establish a baseline for the maximum throughput and minimum time taken by a backup (tape write) of the test data.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration Local Baseline passed.

FCIP Tape Write Acceleration Remote Baseline

The FCIP tape acceleration feature speeds up backups (tape writes) by proxying transfer ready signals and buffering writes at both ends of a possibly very long connection, thereby allowing more than the usual single outstanding write. This test establishes a baseline time for the best possible performance by doing a backup to tape drives from a host on a different fabric connected by FCIP without any added

latency. The time taken for this restore will be the benchmark for later tests using FCIP with and without tape acceleration. This test also helps ensure the tape hardware and backup software are operating correctly.

Test Procedure

The procedure used to perform the FCIP Tape Write Acceleration Remote Baseline test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify tape backup setup so that tape traffic flows over a single FCIP tunnel between a host and tape drives on different fabrics with no added latency and no FCIP tape acceleration enabled. Also ensure filesystems containing the test data are available and tape media and drives in the tape library are available to backup the images. Test data set 1 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server and a Windows 2003 Server . Test data set 2 is Oracle and Exchange database data.

Step 3 Start collecting counters and log files on the MDS switches.

Step 4 Kick off a backup and monitor it for successful completion.

Step 5 After the backup is complete, stop collecting counters and log files on the MDS switches and gather data from both the switches and the Netbackup job status to determine how much data and how many files were backed up in what period of time. Use the "write time" in the job status (not the elapsed time, which includes tape positioning and metadata access).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the backup will succeed and establish a baseline for the maximum throughput and minimum time taken by a backup (tape write) of the test data over an FCIP link without any added latency and without tape acceleration enabled.
- We expect the throughput may be somewhat less than for the local baseline test.
- We expect no CPU or memory problems.

Results

FCIP Tape Write Acceleration Remote Baseline passed.

FSPF Functionality

This section contains the following topics:

- Primary Path Removal VSAN Remove, page 9-116

Basic FSPF Load Balancing

The configuration and verification of load balancing over parallel paths with equal FSPF costs was tested. All configuration and verification was done via the CLI.

Test Procedure

The procedure used to perform the Basic FSPF Load Balancing test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Identify target parallel paths of equal cost in the topology.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across parallel port-channels to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over the available paths.

Step 5 Verify traffic traversing the equal cost parallel paths is evenly distributed across them.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect FSPF to allow for FC traffic to be successfully load balanced over parallel paths of equal cost.
- We expect detection, reporting, and verification to be successfully done with Fabric Manager with CLI confirmation.

Results

Basic FSPF Load Balancing passed.

Path Selection Cost change on Equal Cost Paths

FSPF's capability of changing priority or cost to paths with equal cost was tested. Redundant parallel paths with equal cost were configured so FSPF would select only one as the primary for storage traffic to traverse. All configuration and verification was done via Fabric Manager with confirmation through the CLI.

Test Procedure

The procedure used to perform the Path Selection Cost change on Equal Cost Paths test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Identify target parallel paths of equal cost in the topology.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across parallel port-channels to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over the available paths.

Step 5 Verify that traffic traversing the equal cost parallel paths is evenly distributed across them.

Step 6 Change FSPF cost to one of the parallel equal cost paths to be higher than the other.

Step 7 Confirm traffic changed from load balanced across the parallel paths to only traversing the path with the better metric.

Step 8 Confirm traffic suffered no loss or problems during the path selection configuration.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect FSPF to select a path with better cost than another after cost was changed between parallel and equal paths.
- We expect no traffic loss or problems after the path selection has been configured.
- We expect detection, reporting, and verification to successfully be done through Fabric Manager with CLI confirmation.

Results

Path Selection Cost change on Equal Cost Paths passed.

Primary Path Failure

This test verified FSPF's detection and rerouting of storage traffic after the primary path was shutdown. Traffic was generated over a primary path which was then disabled. All configuration and verification was done via the CLI.

Test Procedure

The procedure used to perform the Primary Path Failure test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Identify target parallel paths of unequal cost in the topology.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across parallel port-channels to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over the primary path.

Step 5 Shut down the primary path link.

Step 6 Confirm the detection of the primary path loss and rerouting of traffic by FSPF over the available redundant path.

Step 7 Confirm traffic suffered no loss or problems during the path selection configuration.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect FSPF to detect the failure (removal) of the primary path and reroute the traffic over a secondary path.
- We expect no traffic loss or problems after the path selection has been configured.
- We expect detection, reporting, and verfication to successfully be done with CLI confirmation.

Results

Primary Path Failure passed.

Primary Path Removal VSAN Remove

This test verified FSPF's detection and rerouting of storage traffic after the VSAN was removed from the primary path's trunk configuration. Traffic was generated over a primary path. The VSAN was removed from the path and rerouting to the secondary path was verified. All configuration and verification was done via Fabric Manager with CLI confirmation.

Test Procedure

The procedure used to perform the Primary Path Removal VSAN Remove test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Identify target parallel paths of unequal cost in the topology.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across parallel port-channels to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over the primary path.

Step 5 Remove the test VSAN from the primary path's allow VSAN list.

Step 6 Confirm the detection of the primary path loss and rerouting of traffic by FSPF over the available redundant path.

Step 7 Confirm storage traffic suffered no loss or problems during the path selection configuration.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect FSPF to detect the removal of a VSAN from the allowed VSAN list of a trunking primary path and rerouting of traffic over a secondary path.
- We expect no traffic loss or problems after the path selection has been configured.
- We expect detection, reporting, validation, verfication was successfully done with CLI confirmation.

Results

Primary Path Removal VSAN Remove passed.

Inter-VSAN Routing Functionality

The Inter-VSAN (IVR) routing functionality tests make sure IVR works as expected both with and without NAT for basic implementations.

This section contains the following topics:

Basic IVR Implementation

Basic inter-VSAN routing (IVR) functionality was tested. Traffic was generated between two edge VSANs via a transit VSAN. Network address translation (NAT) is not used. All configuration was done using Fabric Manager and verification was done via the CLI.

Test Procedure

The procedure used to perform the Basic IVR Implementation test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Create a test topology with two edge switches, each with a different edge VSAN and a common transit VSAN, and a core transit switch. The edge and transit VSANs must have different static domain IDs. Only configure IVR on the edge switches.

Step 3 Configure IVR without NAT on the edge switches to route traffic between the edge VSANs via the transit VSANs.

Step 4 Verify creation and activation of IVR zones. Check that test ports are active in the IVR zone.

Step 5 Generate traffic using the SANtester tool between edge devices. Traffic must use random OXIDs. Verify that storage traffic is delivered without loss or problems to the remote storage array over IVR.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect IVR to successfully route traffic from the source VSAN over a transit VSAN on the transport fabric to a remote destination fabric.
- We expect no loss of traffic or problems with the inter-VSAN routing.

Results

Basic IVR Implementation passed.

Basic IVR NAT Implementation

Basic Inter-VSAN (IVR) routing functionality with Network Address Translation (NAT) was tested between two VSANs with same domain ID. Fiber channel traffic generation was configured to communicate from a source device to a destination device in different VSANs with the same domain ID's. All configuration was done via Fabric Manager with validation through CLI.

Test Procedure

The procedure used to perform the Basic IVR NAT Implementation test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Create a test topology with one edge switch and one core switch (as the border switch), each with the same static domain ID in the edge VSANs.

Step 3 Configure IVR NAT on the border switch to route traffic between the edge VSAN and core VSAN.

Step 4 Verify creation and activation of IVR-NAT zones. Check that test devices are active in the IVR-NAT zones.

Step 5 Generate traffic using the SANtester tool from the edge to the core. Traffic must use random OXIDs.

Step 6 Verify traffic is delivered without loss or problems between the edge and core VSAN ports over IVR-NAT.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect IVR-NAT to successfully route fiber channel traffic from the source VSAN to the destination VSAN when both have the same domain ID.
- We expect no loss of traffic or problems with the Inter-VSAN routing.
- We expect detection, reporting, validation, verfication to be successfully done with CLI confirmation.

Results

Basic IVR NAT Implementation passed.

Port-Channel Functionality

This section contains the following topics:

Basic Port-Channel Load Balancing

This test verified the port channel's load-balancing capability (based on OXID) across all 4 inter-switch link members by generating traffic across the port channel. All configuration and verification was done via the CLI.

Test Procedure

The procedure used to perform the Basic Port-Channel Load Balancing test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 The port channel under test must be configured and active and contain 4 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 4 ISL members to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over the port channel.

Step 5 Verify that storage traffic traversing the port channel is evenly distributed across all 4 members.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic to be evenly distributed across all members of the port channels without loss or problems (based on OXID).
- We expect no CPU or memory problems.

Results

Basic Port-Channel Load Balancing passed.

Multiple Link ADD to Group

This test verified the support for the addition of multiple links to an active port channel group without disrupting active traffic or services over the channel. Traffic was generated to cross the port-channels before, during, and after the links were added. All configuration and verification was done via Fabric Banager with verification through the CLI.

Test Procedure

The procedure used to perform the Multiple Link ADD to Group test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Remove 2 member links from a port channel (via configuration) between an edge and a core switch. The port channel must have 2 ISL members after removal.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across the port channel with 2 ISL members to a core MDS.

Step 4 Verify storage traffic is flowing without loss or problems over port-channels.

Step 5 Verify that storage traffic traversing the 2 ISL port channel members is evenly distributed across all channel members.

Step 6 Add 2 additional ports to the port channel (using the force option).

Step 7 Verify that the newly added port channel members become active in the group. The addition must be detected and reported to the management applications (that is, syslog messages).

Step 8 Verify that storage traffic traversing the port channel is evenly distributed across all 4 members.

Step 9 Confirm that storage traffic traversing the port channel was not affected during or after the addition of the single link to the group.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the multiple link addition to be executed successfully without errors or problems.

- We expect the multiple link addition not to affect active traffic over the port-channel.
- We expect the active storage traffic to be load balanced over the newly added ports.
- We expect no CPU or memory problems.

Results

Multiple Link ADD to Group passed.

Multiple Links Failure in Group

This test verified the physical failure of multiple links in an active port channel group does not disrupt active traffic or services over the channel. Traffic was generated across the port channel before, during, and after the link failure. All configuration and verification was done via the CLI.

Test Procedure

The procedure used to perform the Multiple Links Failure in Group test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 The port channel under test must be configured and active and contain 4 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 4 ISL members to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over the port channel.

Step 5 Verify traffic traversing the port channel is evenly distributed across all 4 members.

Step 6 Physically remove two members of the port channel (disconnect the cables).

Step 7 Verify that the removed port channel members become inactive in the group.

Step 8 Verify that storage traffic traversing the port channel is evenly distributed across all remaining members.

Step 9 Confirm that storage traffic traversing the port channel was not affected during or after the removal of multiple links from the group.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that multiple link failures do not affect active traffic over the port channel beyond some sequence errors at the time the failures occured.
- We expect the failure to be detected and reported by the nodes to the management applications.
- We expect for the active traffic to be load balanced over the remaining port channel members.
- We expect no CPU or memory problems.

Results

Multiple Links Failure in Group passed.

Multiple Links Remove from Group

This test verified the support for the removal (shutdown) of multiple links from an active port channel group without disrupting active traffic or services over the channel. Traffic was generated to cross the port-channel before, during, and after the link was removed. All configuration and verification was done through Fabric Manager with confirmation via the CLI.

Test Procedure

The procedure used to perform the Multiple Links Remove from Group test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 The port channel under test must be configured and active and contain 4 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 4 ISL members to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over the port channel before, during, and after the port removal.

Step 5 Verify traffic traversing the port channel is evenly distributed across all 4 members.

Step 6 Remove 2 members of the port channel.

Step 7 Verify the removed port channel members become inactive in the group.

Step 8 Verify that storage traffic traversing the port channel is evenly distributed across all remaining members.

Step 9 Confirm that traffic traversing the port channel was not affected during or after the removal of the single link from the group.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect multiple link removals (shutdowns) to execute successfully without errors or problems.
- We expect amultiple link removals not to affect active traffic over the port channel.
- We expect the traffic to be load balanced over the remaining port channel members.
- We expect no CPU or memory problems.

Results

Multiple Links Remove from Group passed.

Single Link ADD to Group

This test verified the support for the addition of a single link to an active port channel group without disrupting active traffic or services over the channel. Rraffic was generated to cross the port channel before, during, and after the link was added. All configuration and verification was done using Fabric Manager with verification through the CLI.

Test Procedure

The procedure used to perform the Single Link ADD to Group test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Remove a member link from a port channel (via configuration) between an edge and a core switch. The port channel must have 3 ISL members after removal.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 4 ISL members to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over port channels before, during, and after the port addition.

Step 5 Verify that storage traffic traversing the port channel is evenly distributed across all 3 members.

Step 6 Add a single additional port to the port-channel (using the force option).

Step 7 Verify that the newly added port channel member becomes active in the group. The addition must be detected and reported to the management applications (that is, syslog messages).

Step 8 Verify that storage traffic traversing the port channel is evenly distributed across all 4 members.

Step 9 Confirm that storage traffic traversing the port channel was not affected during or after the addition of the single link to the group.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the single link addition/integration to be executed successfully without errors, issues, or problems.
- We expect the single link addition not to affect active traffic over the port channel.
- We expect the active traffic to be load balanced over the newly added port.

Results

Single Link ADD to Group passed.

Single Link Failure in Group

This test verified the that the physical failure of a single link in an active port channel group does not disrupt active traffic or services over the channel. Traffic was generated across the port channel before, during, and after the link failure. All configuration and verification was done via the CLI.

Test Procedure

The procedure used to perform the Single Link Failure in Group test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 The port channel under test must be configured and active and contain 4 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 4 ISL members to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over the port channel.

Step 5 Verify traffic traversing the port channel is evenly distributed across all 4 members.

Step 6 Physically remove a member of the port channel (disconnect the cable).

Step 7 Verify that the removed port channel member becomes inactive in the group.

Step 8 Verify that storage traffic traversing the port channel is evenly distributed across all remaining members.

Step 9 Confirm that storage traffic traversing the port channel was not affected during or after the removal of the single link from the group except for some sequence errors when the link was removed.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect single link failure not to affect active traffic over the port channel except for some sequence errors when the link fails.
- We expect the failure to be detected and reported by the nodes to the management applications.
- We expect the active traffic to be load balanced over the remaining port channel members.
- We expect no CPU or memory problems.

Results

Single Link Failure in Group passed.

Single Link Remove from Group

This test verified the support for the removal (shutdown) of a single link from an active port channel group without disrupting active traffic or services over the channel. Traffic was generated to cross the port-channel before, during, and after the link was removed. All configuration and verification was done through Fabric Manager with confirmation via the CLI.

Test Procedure

The procedure used to perform the Single Link Remove from Group test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 The port channel under test must be configured and active and contain 4 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 4 ISL members to a core MDS.

Step 4 Verify traffic is flowing without loss or problems over the port channel before, during, and after the port removal.

Step 5 Verify traffic traversing the port channel is evenly distributed across all 4 members.

Step 6 Remove a member of the port channel.

Step 7 Verify that the removed port channel member becomes inactive in the group.

Step 8 Verify that storage traffic traversing the port channel is evenly distributed across all remaining members.

Step 9 Confirm that traffic traversing the port channel was not affected during or after the removal of the single link from the group.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect a single link removal to execute successfully without errors or problems.
- We expect a single link removal not to affect active traffic over the port channel.
- We expect the traffic to be load balanced over the remaining port channel members.
- We expect no CPU or memory problems.

Results

Single Link Remove from Group passed.

Resiliency Functionality

The resiliency functionality tests examine whether single component failures cause unexpected storage access disruption.

This section contains the following topics:

ACTIVE Crossbar Fabric Failover (OIR)

This test verified that a removal/reinsertion of an active crossbar fabric in an edge node causes no disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. An active crossbar fabric module was removed to verify the non disruption of traffic as the other module takes over. The crossbar fabric module was reinserted online to verify full recovery of the failed condition. All configurations and verifications were done via the CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the ACTIVE Crossbar Fabric Failover (OIR) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular switch.

Step 3 Remove an active crossbar fabric from the edge node where storage traffic is entering the fabric. Verify the removal is detected and reported to the management applications.

Step 4 On reinsertion of the module, confirm that it recovers without problems.

Step 5 Verify storage traffic flows without loss or problems.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the removal or reinsertion of the active crossbar fabric because a redundant module is present.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the reinsertion.
- We expect the removal and reinsertion to be detected and reported by the devices to the management application servers (that is, CLI log, Fabric Manager, and SYSLOG server.)
- We expect no CPU or memory problems.

Results

ACTIVE Crossbar Fabric Failover (OIR) passed.

ACTIVE Supervisor Failover (OIR)

This test verified that a removal/reinsertion of the active supervisor in an edge node caused no disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. The active supervisor module was removed to verify the non disruption of traffic as the standby module takes over. The supervisor module was reinserted online (in standby) to verify full recovery of the failed condition. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the ACTIVE Supervisor Failover (OIR) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular switch.

Step 3 Remove the active supervisor from the edge node where storage traffic is entering the fabric.

Step 4 Verify that the standby supervisor becomes active and the reload is detected and reported to the management applications.

Step 5 On reinsertion of the module, confirm that it recovers without problems in standby mode.

Step 6 Verify storage traffic flows without loss or problems.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the removal or reinsertion of the active supervisor because a redundant supervisor modules are present.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the reinsertion.
- We expect the removal and reinsertion to be detected and reported by the devices to the management application servers (that is, CLI log, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

ACTIVE Supervisor Failover (OIR) passed.

ACTIVE Supervisor Failover (Reload)

This test verified that a reload of the active supervisor in an edge switch caused no disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. The active supervisor module was reloaded to verify the non-disruption of traffic. All configurations were done through Fabric Manager and verifications were done via the CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the ACTIVE Supervisor Failover (Reload) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular switch.

Step 3 Execute a reload of the active supervisor to the standby one in an edge node where storage traffic is entering the fabric.

Step 4 Verify that the reload is detected and reported to the management applications.

Step 5 On reload of the module, confirm that it recovers without problems in standby mode.

Step 6 Verify storage traffic flows without loss or problems.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the reload because redundant supervisor modules are present.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the reload.
- We expect the reload to be detected and reported by the devices to the management application servers (that is, CLI log, Fabric Manager and SYSLOG server).
- We expect no CPU or memory problems.

Results

ACTIVE Supervisor Failover (Reload) passed.

ACTIVE Supervisor Failover (manual-CLI)

This test verified that a manual failover of the active supervisor in an edge node causes no disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. The active supervisor module was reloaded to verify the non-disruption of traffic. All configurations were done through CLI and verifications were done via the CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the ACTIVE Supervisor Failover (manual-CLI) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular switch.

Step 3 Execute a manual failover of the active supervisor to the standby one in an edge node where storage traffic is entering the fabric.

Step 4 Verify that the failover is detected and reported to the management applications.

Step 5 On reload of the module, confirm that it recovers without problems in standby mode.

Step 6 Verify storage traffic flows without loss or problems.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the reload because redundant supervisor modules are present.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the reload.
- We expect the reload to be detected and reported by the devices to the management application servers (that is, CLI log, Fabric Manager and SYSLOG server).
- We expect no CPU or memory problems.

Results

ACTIVE Supervisor Failover (manual-CLI) passed.

Back Fan Tray Failure (Removal)

This test verified that the removal of the back fan tray in a core node causes no disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. The back fan tray unit was removed to verify the non disruption of traffic. The fan tray was reinserted and full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Back Fan Tray Failure (Removal) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the storage arrays.

Step 3 Remove the back fan tray unit in a core node where storage traffic is crossing the fabric.

Step 4 Verify that the back fan tray removal is detected and reported to the management applications.

Step 5 Monitor environmental alarms and expect fan tray and possible temperature alarms.

Step 6 Reinsert the back fan tray unit. Confirm that it recovers without problems.

Step 7 Verify storage traffic flows without loss or problems.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be affected by the back fan tray removal if replaced within the specified time (five minutes).
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the removal/reinsertion.
- We expect the back fan tray removal/reinsertion to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Back Fan Tray Failure (Removal) passed.

Core Facing Module Failure (OIR)

This test verified the fabric resiliency to a core facing module removal/reinsertion when port-channeling was running between edge and core nodes with multi module members distribution. Storage traffic (IO's) was generated by the test hosts (Windows and Linux) to the storage array. The core facing module was removed to verify traffic redistribution over remaining port-channel members. The core facing module was reinserted to verify full recovery of the failed condition. All configurations and verifications were done via the fabric manager with confirmation through the CLI. This test verified the fabric resiliency to a core facing module removal/reinsertion on an edge switch. Port-channeling was running between edge and core nodes with members distributed among multiple modules. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. The core facing module was removed and traffic redistribution over remaining port-channel members was verified. The core facing module was reinserted and full recovery from the failed condition was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Core Facing Module Failure (OIR) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular switch.

Step 3 Remove core facing module connecting half of the port-channel member count between the edge node and core node in the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over remaining group members and record any traffic loss.

Step 6 On reinsertion of the module confirm that it recovers without problems.

Step 7 Verify storage traffic flow recovers over the reconnected links.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as the port-channel takes care of distributing it over the remaining group members.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Core Facing Module Failure (OIR) passed.

Core Facing Module Failure (Reload)

This test verified the fabric resiliency to a core facing module reload on an edge switch. Port-channeling was running between edge and core nodes with members distributed among multiple modules. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. The core facing module was reloaded and traffic redistribution over remaining port-channel members was verified. The core facing module was brought back online and full recovery from the failed condition was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Core Facing Module Failure (Reload) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular switch.

Step 3 Reload the core facing module connecting half of the port-channel member count between the edge node and core nodes in the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over remaining group members and record any traffic loss.

Step 6 On reload of the module, confirm that it recovers without problems.

Step 7 Verify storage traffic flow recovers over the reconnected links.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as the port-channel takes care of distributing it over the remaining group members.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Core Facing Module Failure (Reload) passed.

Front FAN TRAY Failure (Removal)

This test verified that the removal of the front fan tray in a core node caused no disruption to the active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. The front fan tray unit was removed and the non-disruption of traffic verified. The fan tray was reinserted after five minutes to ensure the system proactively shut down as designed and to verify full recovery of failed condition after the switch came back up. All configurations and verifications were done via the CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the Front FAN TRAY Failure (Removal) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage arrays.

Step 3 Remove the front fan tray unit in a core node where storage traffic is crossing the fabric.

Step 4 Verify that the front fan tray removal is detected and reported to the management applications.

Step 5 Verify traffic flows are not affected by the removal.

Step 6 Monitor environmental alarms and expect fan tray and possible temperature alarms.

Step 7 Reinsert the front fan tray unit after waiting for more than five minutes and bringing the switch back up. Confirm that it shuts down by itself after five minutes and recovers without problems.

Step 8 Verify storage traffic flows without loss or problems.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be affected by the front fan tray removal if not replaced within the specified time (five minutes).
- We expect connectivity from the test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the removal/reinsertion.
- We expect the front fan tray removal/reinsertion to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Front FAN TRAY Failure (Removal) passed.

Node Failure (Power Loss)

This test verified that a complete core node failure (power loss) caused no disruption to active services and storage traffic beyond the loss of a path for hosts connected to storage arrays connected to the core node. the Storage traffic was generated by the test hosts (Windows and Linux) to storage arrays. One of the core nodes was powered off and the re-routing of traffic through the path on the redundant fabric was verified for the hosts connected to storage on the powered off node. The hosts connected to storage on the other core node were checked to be sure there was no disruption at all. Once the core node was back online, full recovery of the failed condition was verified. All verifications were done via the CLI and Fabric Manager.

Test Procedure

The procedure used to perform the Node Failure (Power Loss) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the storage arrays.

Step 3 Power down one of the core nodes where storage traffic is crossing the fabric, then power it back up after about 15-30 seconds.

Step 4 Verify that the core node loss is detected and reported to the management applications.

Step 5 Verify traffic flows are not affected by the core node loss beyond the loss of a path for the hosts connected to storage ports on the core node.

Step 6 Confirm that it recovers without problems.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect host I/O traffic not to be stopped by the complete loss and recovery of a core node because a redundant fabric is present.
- We expect host I/O traffic not to be affected on the redundant core node in the same fabric as the failed core node.
- We expect systems to recover completely from the core node loss/recovery.
- We expect the core node loss to be detected and reported by the devices to the management servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Node Failure (Power Loss) passed.

Node Failure (Reload)

This test verified that a complete core node failure (reload) caused no disruption to active services and storage traffic beyond the loss of a path for hosts connected to storage arrays connected to the core node. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. One of the core nodes was reloaded and the re-routing of traffic through the path on the redundant fabric was verified for the hosts connected to storage on the powered off node. The hosts connected to storage on the other core node were checked to be sure there was no disruption at all. Once the core node was back online, full recovery of the failed condition was verified. All configuration was done via Fabric Manager and verifications were done via the CLI and Fabric Manager.

Test Procedure

The procedure used to perform the Node Failure (Reload) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the storage arrays.

Step 3 Reload one of the core nodes where storage traffic is crossing the fabric.

Step 4 Verify that the core node loss is detected and reported to the management applications.

Step 5 Verify traffic flows are not affected by the core node loss beyond the loss of a path for the hosts connected to storage ports on the core node.

Step 6 Once online, confirm that it recovers without problems.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect host I/O traffic not to be stopped by the complete loss and recovery of a core node because a redundant fabric is present.
- We expect host I/O traffic not to be affected on the redundant core node in the same fabric as the failed core node.
- We expect systems to recover completely from the core node reload.
- We expect the core node reload to be detected and reported by the devices to the management servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Node Failure (Reload) passed.

Power Supply Failure (Cord Removal)

This test verified that a loss of a power supply unit due to power cords removal in a core node caused no disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The power cord from one of the two power supply units was removed and the non-disruption of traffic was verified. The power cord was reconnected and full recovery of the failed condition was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Power Supply Failure (Cord Removal) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the storage arrays.

Step 3 Unplug the power cable off one of the power supply units in a core node where storage traffic is crossing the fabric.

Step 4 Verify that the power supply loss of power is detected and reported to the management applications.

Step 5 Plug the power supply and confirm that it recovers without problems.

Step 6 Verify traffic flows are not affected by the power loss or recovery.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the power supply loss because a redundant power supply unit is present.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the power loss/recovery.
- We expect the power supply power loss to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Power Supply Failure (Cord Removal) passed.

Power Supply Failure (PowerOff)

This test verified that a loss of a power supply unit in a core node caused no disruption to active services and storage traffic. Storage traffic were generated by the test hosts (Windows and Linux) to the storage arrays. One of the two power supply units was powered off and the non-disruption of traffic was verified. The power supply was powered back on and full recovery of the failed condition was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Power Supply Failure (PowerOff) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the storage arrays.

Step 3 Power down one of the power supply units in a core node where storage traffic is crossing the fabric.

Step 4 Verify that the power supply shut down is detected and reported to the management applications.

Step 5 Power the unit and confirm that it recovers without problems.

Step 6 Verify traffic flows are not affected by the power loss or recovery.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the power supply loss because a redundant power supply unit is present.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the power loss/recovery.
- We expect the power supply shut off to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Power Supply Failure (PowerOff) passed.

Power Supply Failure (Removal)

This test verified that the removal of a power supply unit in a core node caused no disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. One of the two power supply units was removed and the non-disruption of traffic was verified. The power supply was reinserted and full recovery of the failed condition was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Power Supply Failure (Removal) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the storage arrays.

Step 3 Remove one of the power supply units in a core node where storage traffic is crossing the fabric.

Step 4 Verify that the power supply removal is detected and reported to the management applications.

Step 5 Reinsert the power supply and power the unit. Confirm that it recovers without problems.

Step 6 Verify traffic flows are not affected by the power loss or recovery.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the power supply loss because a redundant power supply unit is present.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the power loss/recovery.
- We expect the power supply removal/reinsertion to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Power Supply Failure (Removal) passed.

SAN OS Code Upgrade Event

This test verified that a code upgrade to a core node causes NO disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays, and synthetic traffic at line rate was generated between two ports on different linecards using an Agilent SAN Tester. Core node was upgraded to verify the ’hitless upgrade’ non-disruption of traffic. All configurations and verifications were done via CLI with confirmation through Fabric Manager and CLI.

Test Procedure

The procedure used to perform the SAN OS Code Upgrade Event test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test-hosts (OS: Windows and Linux) to the storage arrays.

Step 3 Generate IO using SAN Tester at line rate to two different line cards.

Step 4 Upgrade the SANOS code in a core node where storage traffic is entering the fabric.

Step 5 Verify that the upgrade procedure is detected and reported to the management applications.

Step 6 Once the upgrade is completed confirm that the node is without problems.

Step 7 Verify storage traffic flows without loss or problems.

Step 8 Verify SAN tester traffic flows without loss or problems.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that traffic is not stopped by the upgrade of the core node.
- We expect connectivity from test hosts to arrays is not affected by the upgrade.
- We expect all systems to recover completely from the procedure.
- We expect the upgrade to be detected and reported by the devices to the management application servers (e.g. Fabric Manager, SYSLOG, etc.)
- We expect no CPU or memory problems.

Results

SAN OS Code Upgrade Event passed.

STANDBY Supervisor Failure (OIR)

This test verified that a removal/reinsertion of the standby supervisor in an edge node caused no disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. The standby supervisor module was removed to verify the non disruption of traffic. The supervisor module was reinserted and came up online (in standby) to verify full recovery of the failed condition. All configurations and verifications were done via the CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the STANDBY Supervisor Failure (OIR) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular switch.

Step 3 Remove the standby supervisor in an edge node where storage traffic is entering the fabric, then after a minute or so reinsert it.

Step 4 Verify that the removal and reinsertion are detected and reported to the management applications.

Step 5 Verify traffic flows are not affected by the standby supervisor removal.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the removal because the supervisor module is the standby unit.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the removal/reinsertion.
- We expect the removal/reinsertion to be detected and reported by the devices to the management application servers (that is, CLI log, Fabric Manager, and SYSLOG server.)
- We expect no CPU or memory problems.

Results

STANDBY Supervisor Failure (OIR) passed with exception. The following exceptions were noted: CSCsk96269.

STANDBY Supervisor Failure (Reload)

This test verified that a reload of the standby supervisor in an edge node caused no disruption to active services and storage traffic. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. The standby supervisor module was reloaded to verify the non disruption of traffic and that the supervisor module came back online (in standby). All configurations and verifications were done via the CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the STANDBY Supervisor Failure (Reload) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular switch.

Step 3 Execute a reload of the standby supervisor in an edge node where storage traffic is entering the fabric.

Step 4 Verify that the reload is detected and reported to the management applications.

Step 5 On reload of the module, confirm that it recovers without problems in standby mode.

Step 6 Verify storage traffic flows without loss or problems.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the reload of the standby supervisor module.
- We expect connectivity from test hosts to arrays not to be affected by the failure or recovery.
- We expect systems to recover completely from the reload.
- We expect the reload to be detected and reported by the devices to the management application servers (that is, CLI log, Fabric Manager, and SYSLOG server.)
- We expect no CPU or memory problems.

Results

STANDBY Supervisor Failure (Reload) passed.

Unused Module Failure (OIR)

This test verified the fabric resiliency to the removal and insertion of an unused module in an edge node with storage traffic running through it. Storage traffic was generated by the test hosts (Windows and Linux) to the storage arrays. An unused module in the edge node was removed and then reinserted to verify that there is no effect on storage traffic. All configurations and verifications were done via the CLI with confirmation through Fabric manager.

Test Procedure

The procedure used to perform the Unused Module Failure (OIR) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the switch.

Step 3 Remove the unused edge node module unrelated to the test hosts or core connections.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 On reinsertion of the module, confirm that it recovers without problems.

Step 6 Verify storage traffic flow is not affected by the reinsertion.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect connectivity from test hosts to arrays not to be affected by the removal and insertion of an unused module.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Unused Module Failure (OIR) passed.

EMC Clariion

This section contains the following topics:

Host Facing Module Failure (OIR) EMC CLARiiON

This test verified the fabric and host resiliency to a host-facing module removal and reinsertion when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Facing Module Failure (OIR) EMC CLARiiON test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Remove the edge module(s) connecting the test hosts and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over redundant path.

Step 6 Reinsert the module(s) and confirm recovery takes place without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.

- We expect systems to recover completely from the module removal and reinsertion.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Facing Module Failure (OIR) EMC CLARiiON passed.

Host Facing Module Failure (Reload) EMC CLARiiON

This test verified the fabric and host resiliency to a host-facing module reload when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Facing Module Failure (Reload) EMC CLARiiON test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Reload the edge module(s) connecting the test hosts and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over redundant path.

Step 6 On reload of the module(s) confirm recovery takes place without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Facing Module Failure (Reload) EMC CLARiiON passed.

Host Link Failure (Link pull)—EMC CLARiiON

This test verified the fabric and host resiliency to a link failure due to a cable disconnection when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. When the host port was reconnected, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Link Failure (Link pull)—EMC CLARiiON test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Physically remove a link between the test hosts and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over the redundant path.

Step 6 Reconnect the links and confirm they recover without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the link failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Link Failure (Link pull)—EMC CLARiiON passed.

Host Link Failure (Port Shutdown) EMC CLARiiON

This test verified the fabric and host resiliency to a port shutdown when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. One host facing port from each test host was disabled to verify traffic rerouted to redundant connections. When the host port was enabled, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Link Failure (Port Shutdown) EMC CLARiiON test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Shut down a switch link for each of the test hosts.

Step 4 Verify that the shut down is detected and reported to the management applications.

Step 5 Verify traffic flows completely over the redundant path.

Step 6 Reenable the link and confirm that it recovers without problems.

Step 7 Verify storage traffic flow recovers over the reenabled link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the link failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the host port resets.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Link Failure (Port Shutdown) EMC CLARiiON passed.

EMC DMX

The EMC resiliency functionality tests involve failing a path by disabling an edge link, disconnecting an edge link, and reloading and replacing an edge switch module and making sure the fabric design and the PowerPath multipath software transparently route all traffic over the remaining link.

This section contains the following topics:

Host Link Failure Link Pull EMC DMX

This test verified the fabric and host resiliency to a link failure due to a cable disconnection when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Link Failure Link Pull EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Physically remove a link between the test host and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over the redundant path.

Step 6 Reconnect the links and confirm they recover without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the link failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Link Failure Link Pull EMC DMX passed.

Host Link Failure Port Shutdown EMC DMX

This test verified the fabric and host resiliency to a port shutdown when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Link Failure Port Shutdown EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Shut down a switch link for each of the test hosts.

Step 4 Verify that the shut down is detected and reported to the management applications.

Step 5 Verify traffic flows completely over the redundant path.

Step 6 Reenable the link and confirm that it recovers without problems.

Step 7 Verify storage traffic flow recovers over the reenabled link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the link failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Link Failure Port Shutdown EMC DMX passed.

Host Module Failure OIR EMC DMX

This test verified the fabric and host resiliency to a host-facing module removal and reinsertion when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Module Failure OIR EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Remove the edge module(s) connecting the test hosts and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over redundant path.

Step 6 Reinsert the module(s) and confirm recovery takes place without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module removal and reinsertion.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Module Failure OIR EMC DMX passed.

Host Module Failure Reload EMC DMX

This test verified the fabric and host resiliency to a host-facing module reload when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Module Failure Reload EMC DMX test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Reload the edge module(s) connecting the test hosts and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over redundant path.

Step 6 On reload of the module(s) confirm recovery takes place without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Module Failure Reload EMC DMX passed.

HP XP

The HP resiliency functionality tests involve failing a path by disabling an edge link, disconnecting an edge link, and reloading and replacing an edge switch module and making sure the fabric design and host multipath software (native MPIO for Linux, HP MPIO-DSM for Windows) transparently route all traffic over the remaining link

This section contains the following topics:

Host Link Failure Link Pull HP XP

This test verified the fabric and host resiliency to a link failure due to a cable disconnection when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Link Failure Link Pull HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Physically remove a link between the test host and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over the redundant path.

Step 6 Reconnect the links and confirm they recover without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the link failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.

- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Link Failure Link Pull HP XP passed.

Host Link Failure Port Shutdown HP XP

This test verified the fabric and host resiliency to a port shutdown when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Link Failure Port Shutdown HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Shut down a switch link for each of the test hosts.

Step 4 Verify that the shut down is detected and reported to the management applications.

Step 5 Verify traffic flows completely over the redundant path.

Step 6 Reenable the link and confirm that it recovers without problems.

Step 7 Verify storage traffic flow recovers over the reenabled link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the link failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Link Failure Port Shutdown HP XP passed.

Host Module Failure OIR HP XP

This test verified the fabric and host resiliency to a host-facing module removal and reinsertion when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Module Failure OIR HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Remove the edge module(s) connecting the test hosts and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over redundant path.

Step 6 Reinsert the module(s) and confirm recovery takes place without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module removal and reinsertion.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Module Failure OIR HP XP passed.

Host Module Failure Reload HP XP

This test verified the fabric and host resiliency to a host-facing module reload when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Module Failure Reload HP XP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Reload the edge module(s) connecting the test hosts and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over redundant path.

Step 6 On reload of the module(s) confirm recovery takes place without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Module Failure Reload HP XP passed.

NetApp

The NetApp resiliency functionality tests involve failing a path by disabling an edge link, disconnecting an edge link, and reloading and replacing an edge switch module and making sure the fabric design and host multipath software (native MPIO for Linux, ONTAP DSM for Windows) transparently route all traffic over the remaining link

This section contains the following topics:

Host Link Failure Link Pull NetApp

This test verified the fabric and host resiliency to a link failure due to a cable disconnection when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Link Failure Link Pull NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Physically remove a link between the test host and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over the redundant path.

Step 6 Reconnect the links and confirm they recover without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the link failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.

- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Link Failure Link Pull NetApp passed.

Host Link Failure Port Shutdown NetApp

This test verified the fabric and host resiliency to a port shutdown when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Link Failure Port Shutdown NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Shut down a switch link for each of the test hosts.

Step 4 Verify that the shut down is detected and reported to the management applications.

Step 5 Verify traffic flows completely over the redundant path.

Step 6 Reenable the link and confirm that it recovers without problems.

Step 7 Verify storage traffic flow recovers over the reenabled link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the link failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Link Failure Port Shutdown NetApp passed.

Host Module Failure OIR NetApp

This test verified the fabric and host resiliency to a host-facing module removal and reinsertion when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Module Failure OIR NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Remove the edge module(s) connecting the test hosts and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over redundant path.

Step 6 Reinsert the module(s) and confirm recovery takes place without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module removal and reinsertion.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Module Failure OIR NetApp passed.

Host Module Failure Reload NetApp

This test verified the fabric and host resiliency to a host-facing module reload when multi-pathing software was running in a host with redundant paths. Storage traffic was generated by the test hosts (Windows and Linux) to the storage array. The test host facing module was reloaded to verify traffic rerouted to redundant connections. When the edge module came back online, full recovery was verified. All configurations and verifications were done via the CLI with confirmation through the Fabric Manager.

Test Procedure

The procedure used to perform the Host Module Failure Reload NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Generate storage traffic from multiple test hosts (Windows and Linux) to the particular storage array.

Step 3 Reload the edge module(s) connecting the test hosts and the fabric.

Step 4 Verify that the failure is detected and reported to the management applications.

Step 5 Verify traffic flows completely over redundant path.

Step 6 On reload of the module(s) confirm recovery takes place without problems.

Step 7 Verify storage traffic flow recovers over the reconnected link.

Step 8 Verify link recovery is detected and reported by the devices to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect traffic not to be stopped by the failure as multi-pathing software takes care of failing it over to the redundant host link.
- We expect traffic loss to be null or minimal during failure and recovery.
- We expect systems to recover completely from the module reload.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (that is, Fabric Manager and SYSLOG server.)
- We expect no CPU or memory problems.

Results

Host Module Failure Reload NetApp passed.

Security Functionality

This section contains the following topics:

FC SP Authentication Failure

This test verifies FC-SP's capability to reject an unauthorized node from joining the fabric. All configuration and verification was done via Fabric Manager with confirmation through CLI.

Test Procedure

The procedure used to perform the FC SP Authentication Failure test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Configure a fabric with FS-SP.

Step 3 Verify the configuration and successful authorization of all nodes in the fabric. All members (all fabric nodes) must be active. The testbed must be fully operational.

Step 4 Change an edge MDS FC-SP configuration to have the wrong key. Try to reconnect to the fabric.

Step 5 Verify that the edge MDS is rejected (prohibited from joining the fabric).

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect FC-SP to successfully reject the integration of an edge switch with wrong credentials.
- We expect detection, reporting, and verfication to be successfully done with the Fabric Manager with CLI confirmation.
- We expect no CPU or memory problems.

Results

FC SP Authentication Failure passed.

Port Security Basic Implementation

The configuration and verification of port security was tested. A single host was allowed access and then replaced with another non-authorized host. All configuration and verification was done via Fabric Manager with confirmation through the CLI.

Test Procedure

The procedure used to perform the Port Security Basic Implementation test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Configure port security for auto learning in a single VSAN on a fabric.

Step 3 Verify the test host port is learned by port security.

Step 4 Disable port-security auto learning mode.

Step 5 Generate storage traffic from a SANtester port on an edge switch to a core switch.

Step 6 Verify storage traffic is flowing without loss or problems across the fabric.

Step 7 Change the end port PWWN and verify that port security rejects the new connection at FLOGI time because it is not allowed.

Step 8 Verify that port security detects and rejects the wrong host connection and reports it to the management applications.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect port security to be configured without issues.
- We expect port security to automatically learn the address of the test host and keep it locked after auto learn is disabled.
- We expect port security to not allow a non-authorized node to login into the fabric.
- We expect that detection, reporting, validation, verfication was successfully done by Fabric Manager with CLI confirmation.

Results

Port Security Basic Implementation passed.

User Access TACACS Basic Test

This test verified TACACS+ support in the MDS as the primary authorization/authentication mechanism in the testbed. Remote TACACS+ authorization/authentication was validated. All configuration and verification was done via the Fabric Manager with confirmation through the CLI.

Test Procedure

The procedure used to perform the User Access TACACS Basic Test test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Remote TACACS+ authorization/authentication must be configured and services active and available.

Step 3 Access the target node and login using the remote (TACACS+) username/password configured in the TACACS+ server configuration. Use a username which is not configured in Fabric Manager for discovery.

Step 4 Verify that the access and administrator authorization is granted then logout.

Step 5 Access the target node and login using the local username/password configured in the nodes configuration. Verify that access is not granted (that is, access fails with local username/password combination).

Step 6 Access the target node and login using the wrong username/password combinations. Verify that access is not granted.

Step 7 Verify that all rejections are reported to the management applications.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect successful access and remote TACACS+ authorization/authentication.
- We expect access to be denied with wrong username/password combination.
- We expect detection, reporting, validation, and verfication to successfully be done with the Fabric Manager with the CLI confirmation.
- We expect no CPU or memory problems.

Results

User Access TACACS Basic Test passed.

User Access TACACS Servers Failure

This test verified TACACS+ support for redundant servers and local authentication as a last resort in the MDS. All configuration and verification was done via the Fabric Manager with confirmation through the CLI.

Test Procedure

The procedure used to perform the User Access TACACS Servers Failure test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Remote (TACACS+) authorization/authentication must be configured and services active/available.

Step 3 Access the target node and log in using the remote (TACACS+) username/password configured in the TACACS+ server configuration. Use a username which is not configured in Fabric Manager for discovery.

Step 4 Verify that access and administrator authorization is granted then log out.

Step 5 Take primary TACACS+ server offline and attempt to log in again with the predefined username/password.

Step 6 Verify that access and administrator authorization is granted using the second TACACS+ server. Confirm via the CLI that the primary TACACS+ server is offline.

Step 7 Take secondary TACACS+ server off line and attempt to log in again with the predefined username/password. Verify that access failed using TACACS+ defined username/password.

Step 8 Attempt to log in using local authentication username/password. Once logged in verify that both TACACS+ servers are down.

Step 9 Bring both TACACS+ servers online and attempt to login through them. Verify full connectivity from the target MDS to the TACACS+ servers.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the MDS to successfully access the TACACS+ secondary server if communication to the primary is lost.
- We expect local authorization to be used as last resort when all TACACS+ servers are down.
- We expect detection, reporting, validation, and verfication to successfully be done with the Fabric Manager with CLI confirmation.
- We expect no CPU or memory problems.

Results

User Access TACACS Servers Failure passed.

Zone Scalability

Zone scalability tests ensure the maximum supported number of zones and zone members can be configured.

This section contains the following topics:

Maximum Zone Members (Basic Zoning with Device Alias)

The maximum number of zone members per physical fabric was tested. Members were zoned using the device alias. Configuration guidelines are taken from "Configuration Limits for Cisco MDS SAN-OS Release 3.x" (http://www.cisco.com/en/US/partner/products/ps5989/products_configuration_guide_chapter09186a0080679bc9.html). All configuration was done using the CLI or Fabric Manager as noted in the step descriptions and verification was done via the CLI.

Test Procedure

The procedure used to perform the Maximum Zone Members (Basic Zoning with Device Alias) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Prior to the test, create a topology with a fabric consisting of two edge and two core switches, each with a common VSAN trunked on the appropriate ISLs. Connect 2 Agilent SAN Tester ports to each switch in the VSAN. Be sure basic zoning is in use, the default zone policy is deny, and full zoneset transfer is in effect.

Step 3 Configure enough zones using device aliases to reach the maximum number of zone members, add them to a zoneset, and activate the zoneset using the CLI, then verify creation and activation of the zoneset and check that test ports are active in the zoneset and can pass traffic.

Step 4 Using Fabric Manager, remove a zone member, reactivate the zoneset, and check that test ports are active in the zoneset and can pass traffic.

Step 5 Using Fabric Manager, add a zone member, reactivate the zoneset, and check that test ports are active in the zoneset and can pass traffic.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect zone creation and modification (deletion and addition of a single zone) and zoneset activation to succeed and allow frames to pass between zone members when configured with the maximum number of zones per physical fabric.
- We expect no loss of traffic or problems with the basic zoning.
- We expect no CPU or memory problems.

Results

Maximum Zone Members (Basic Zoning with Device Alias) passed.

Maximum Zone Members (Basic Zoning with PWWN)

The maximum number of zone members per physical fabric was tested. Members were zoned using the PWWN. Configuration guidelines are taken from "Configuration Limits for Cisco MDS SAN-OS Release 3.x" (http://www.cisco.com/en/US/partner/products/ps5989/products_configuration_guide_chapter09186a0080679bc9.html). All configuration was done using the CLI or Fabric Manager as noted in the step descriptions and verification was done via the CLI.

Test Procedure

The procedure used to perform the Maximum Zone Members (Basic Zoning with PWWN) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Prior to the test, create a topology with a fabric consisting of two edge and two core switches, each with a common VSAN trunked on the appropriate ISLs. Connect 2 Agilent SAN Tester ports to each switch in the VSAN. Be sure basic zoning is in use, the default zone policy is deny, and full zoneset transfer is in effect.

Step 3 Configure enough zones using PWWNs to reach the maximum number of zone members, add them to a zoneset, and activate the zoneset using the CLI, then verify creation and activation of the zoneset and check that test ports are active in the zoneset and can pass traffic.

Step 4 Using Fabric Manager, remove a zone member, reactivate the zoneset, and check that test ports are active in the zoneset and can pass traffic.

Step 5 Using Fabric Manager, add a zone member, reactivate the zoneset, and check that test ports are active in the zoneset and can pass traffic.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect zone creation and modification (deletion and addition of a single zone) and zoneset activation to succeed and allow frames to pass between zone members when configured with the maximum number of zones per physical fabric.
- We expect no loss of traffic or problems with the basic zoning.
- We expect no CPU or memory problems.

Results

Maximum Zone Members (Basic Zoning with PWWN) passed.

CHAPTER 10

Wide Area Application Services (WAAS) ACE

Cisco Wide Area Application Services 4.0 (WAAS) is a powerful application acceleration and WAN optimization solution for the branch office that improves the performance of any TCP-based application operating in a Wide Area Network (WAN) environment. The WAAS software is built on the WAFS framework and still provides WAFS functionality as well as some added optimization features.

With Cisco WAAS, enterprises can consolidate costly branch office servers and storage into centrally managed data centers, while still offering LAN-like service levels for remote users.

The solution offers a significantly lower total cost of ownership (TCO), greater application performance, more efficient WAN usage, and transparent integration with the network with secure, centralized manageability and control in an easy-to-implement package. Cisco WAAS provides the technologies necessary to enable consolidation of infrastructure into the data center while also providing application acceleration and Wide Area Network (WAN) optimization capabilities that achieve application delivery performance similar to that of a Local Area Network (LAN).

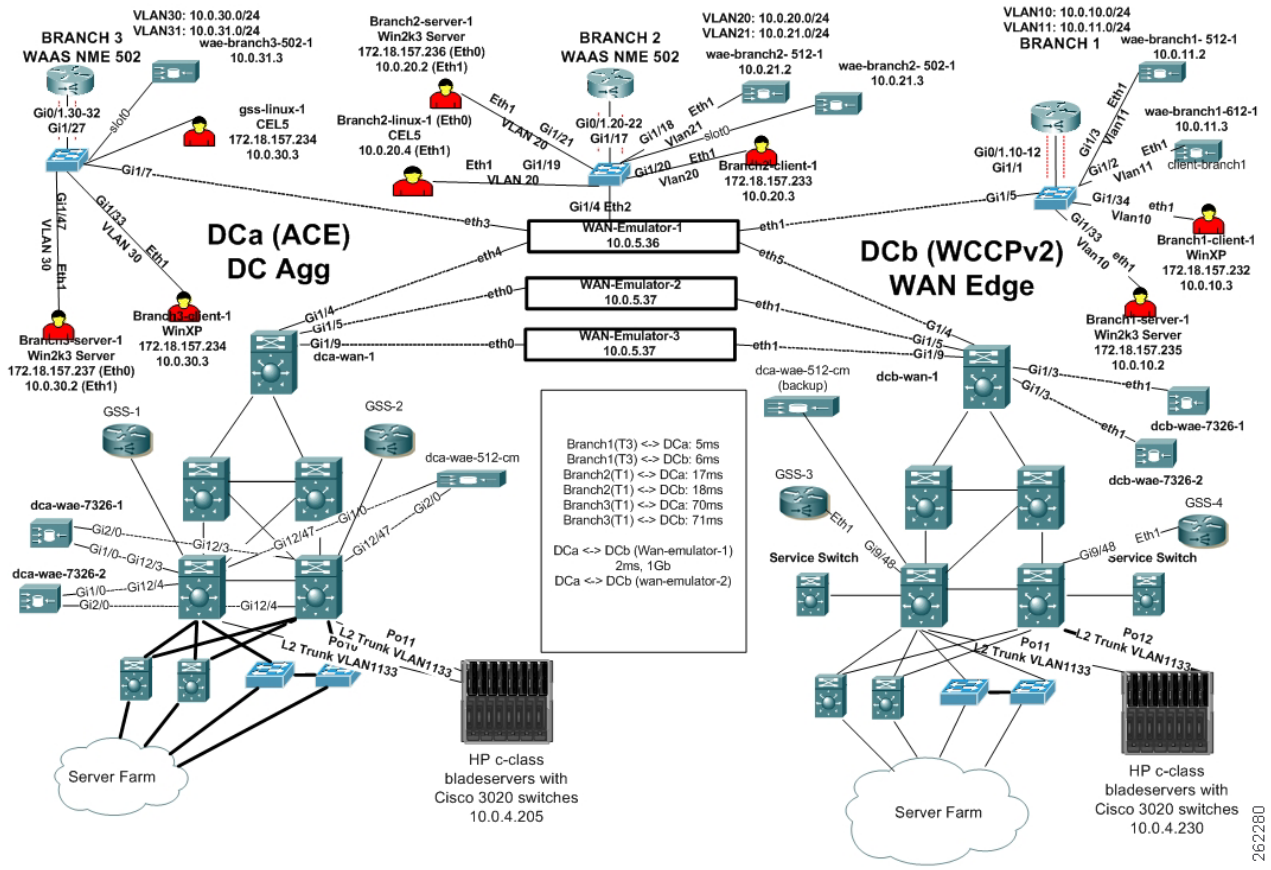
The solution provides the LAN-like performance across the WAN through a combination of technologies, including:

- Application acceleration—Mitigate latency and bandwidth through advanced protocol optimizations, including read-ahead, message prediction, and caching.
- Throughput optimization—Improve behavior of transport protocols to make them more efficient in WAN environments.
- Bandwidth optimization—Minimize the transmission of redundant data patterns through data redundancy elimination (DRE) and compression.

WAAS Topology

Cisco WAAS software running on Cisco Wide Area Application Engine (WAE) platforms is deployed in the data center and remote office locations as appliances attached to the LAN or as network modules (NME-WAE) integrated with the branch router. Cisco WAAS employs the Web Cache Communication Protocol (WCCP) v2, Policy-Based Routing (PBR), or server load balancers suce as the ACE to intercept traffic and transparently forward it to the local Cisco WAE on both sides of the network (Figure 10-1).

Figure 10-1 DCAP WAAS Test Topology



Test Results Summary

Table 10-1 on page 10-3 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 10-1 on page 10-3 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTS's.

A number of resources were referenced during the design and testing phases of Cisco WAAS in DCAP. These include the WAAS Design Guide, produced by Cisco's Enterprise Solution Engineering Data Center team, and Cisco's WAAS Maintenance Guide. Find links to these two documents below.

Enterprise Data Center Wide Area Application Services (WAAS) Design Guide (SRND):

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns377/c649/ccmigration_09186a008081c7da.pdf

Cisco Wide Area Application Services Configuration Guide, Chapter 14: Maintaining Your WAAS System (Maintenance):

http://www.cisco.com/univercd/cc/td/doc/product/webscale/waas/waas407/cfgd/maint.htm

Table 10-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Acceleration, page 10-4	n/a	1. ACE Redirection HTTP Acceleration All Branches 2. ACE Redirection FTP Acceleration All Branches	
CIFS, page 10-7	n/a	1. Cache Miss Benchmark ACE Redirection	
Redirection, page 10-8	n/a	1. ACE WAAS Configuration and Verification	
WAFS, page 10-10	n/a	1. Cache Hit Benchmark ACE Redirection	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Acceleration

Cisco Wide Area Application Services 4.0 (WAAS) is a powerful application acceleration and WAN optimization solution for the branch office that improves the performance of any TCP-based application operating in a Wide Area Network (WAN) environment. The WAAS software is built on the WAFS framework and still provides WAFS functionality as well as some added optimization features. With WAAS WAN traffic is optimized in three different ways, TCP Flow Optimization (TFO), Data Redundancy Elimination (DRE), and LZ Compression.

This section contains the following topics:

ACE Redirection HTTP Acceleration All Branches

This test verified the ability of WAAS to accelerate HTTP traffic when using ACE for traffic redirection. From each branch, 1 client is simulated initiating an HTTP session to a server in the Data Center. The client makes a request for a 500k URL resource. There is one HTTP GET per connection and a 10ms delay between sessions. The traffic was run for 3 minutes from each branch to a simulated FTP server in the Data Center both with and without optimization.

It was verified and quantified that HTTP response times were improved when WAAS was enabled compared to native WAN response times. ACE redirection was used at the core of the WAAS network in the Data Center. Traffic was first run unoptimized by turning off WCCPv2 redirection at each branch. The same traffic was then run, optimized, by turning on WCCPv2 redirection at each branch. ACE was configured for TCP redirection to the core WAE's throughout the entire test.

Test Procedure

The procedure used to perform the ACE Redirection HTTP Acceleration All Branches test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On each branch router disable WCCPv2 redirects with the **no ip wccp 61** and **no ip wccp 62** commands. This will cause all traffic to be unoptimized by WAAS.

Step 3 With the Shenick Test tool begin a traffic stream. The stream should be set up to simulate 1 user at each branch each making 1 HTTP GET request for a 100k URL resource every 10ms for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 4 Analyze the results and report the average transaction response time for each branch.

Step 5 On the branch routers enable WCCPv2 redirection with the **ip wccp 61** and **ip wccp 62** commands. Verify WCCPv2 state becomes usable with the **show ip wccp 61 detail** and **show ip wccp 62 detail** commands.

Step 6 Clear the statistics on the core WAEs by issuing the **clear statistics all** command.

Step 7 With the Shenick Test tool begin a traffic stream. The stream should be set up to simulate 1 user at each branch each making 1 HTTP GET request for a 100k URL resource every 10ms for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 8 While the traffic is running verify on each core WAE that the connections are being load balanced and optimized by issuing the **show tfo connection summary** command.

Step 9 Analyze the results and report the average transaction response time for each branch.

Step 10 Collect statistics on each core WAE devices by issuing the following commands:

```
  show statistics tfo  show statistics tfo saving  show statistics tcp  show statistics
dre  Verify that TFO savings were seen for inbound and outbound web based traffic.
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that HTTP traffic will pass when WAAS is enabled.
- We expect that HTTP GET response times will decrease with WAAS acceleration enabled.
- We expect ACE to load-balance connections across each WAE.

Results

ACE Redirection HTTP Acceleration All Branches passed.

ACE Redirection FTP Acceleration All Branches

This test verified the ability of WAAS to accelerate FTP traffic when using ACE for traffic redirection. From each branch, 1 client is simulated initiating an FTP session to a server in the Data Center, and issuing the bin, pwd, cd, ls, and finally get commands. The client makes a request for a file size of 1MB. There is a 10ms delay between FTP sessions. The traffic was run for 3 minutes from each branch to a simulated FTP server in the Data Center both with and without optimization.

It was verified and quantified that FTP response times were improved when WAAS was enabled compared to native WAN response times. ACE redirection was used at the core of the WAAS network in the Data Center. Traffic was first run unoptimized by turning off WCCPv2 redirection at each branch. The same traffic was then run, optimized, by turning on WCCPv2 redirection at each branch. ACE was configured for TCP redirection to the core WAE's throughout the entire test.

Test Procedure

The procedure used to perform the ACE Redirection FTP Acceleration All Branches test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On each branch router disable WCCPv2 redirects with the **no ip wccp 61** and **no ip wccp 62** commands. This will cause all traffic to be unoptimized by WAAS.

Step 3 With the Shenick Test tool begin a traffic stream. The stream should be set up to simulate 1 users each making 1 FTP GET request for a 1MB file every 10ms for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 4 Analyze the results and report the average transaction response time for each branch.

Step 5 On the branch router enable WCCPv2 redirection with the **ip wccp 61** and **ip wccp 62** commands. Verify WCCPv2 State becomes usable with the **show ip wccp 61 detail** and **show ip wccp 62 detail** commands.

Step 6 Clear the statistics on the WAE at the branch by issuing the **clear statistics all** command.

Step 7 With the Shenick Test tool begin a traffic stream. The stream should be set up to simulate 1 users each making 1 FTP GET request for a 1MB file every 10ms for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 8 While the traffic is running verify WAE that the connections are being optimized by issuing the **show tfo connection summary** command.

Step 9 Analyze the results and report the average transaction response time for each branch. Compare with the statistics collected previously when WAAS was not optimizing traffic.

Step 10 Collect statistics on the WAE devices at the branch by issuing the following commands:

```
  show statistics tfo  show statistics tfo saving  show statistics tcp  show statistics
dre  Verify that TFO savings were seen for inbound and outbound Web based traffic. Also
verify that DRE savings were measured appropriately.
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that FTP traffic will pass when WAAS is enabled.
- We expect that FTP GET response times will decrease with WAAS acceleration enabled.

Results

ACE Redirection FTP Acceleration All Branches passed.

CIFS

Common Internet File System (CIFS) is a protocol that defines a standard for remote file access. With CIFS, users with different platforms and computers can share files without having to install new software. CIFS runs over TCP/IP but uses the SMB (Server Message Block) protocol found in Microsoft Windows for file and printer access; therefore, CIFS will allow all applications, not just Web browsers, to open and share files across the Internet. With CIFS, changes made to a file are simultaneously saved on both the client and server side. Clients in a WAAS network use the CIFS cache service to request file and print services from servers over a network. The tests performed verified the functionality and baseline performance for CIFS with, and without, the WAAS software.

This section contains the following topics:

Cache Miss Benchmark ACE Redirection

The Cisco WAFS Benchmark Tool copies workload files ranging from 50k to 2MB MS Word, Excel, and Powerpoint to the a server and then begins the test. Each file on the server is opened by the client, modified, saved, and closed. Times are recorded for each operation. The cache miss, or cold cache benchmark, tests tests how quickly a file server can be accessed for the first time (with no data in the WAFS cache).

In this test, the WAFS Benchmark tool was run while WCCPv2 redirection at the branch and ACE redirection at the Data Center was enabled. The edge WAE cache was first cleared to simulate a clear WAFS cache, and then the Benchmark tool was started. The output of the benchmark tool provided performance results for files that were not yet cached locally. Each transaction was optimzed with TFO and DRE causing the overall transaction response time to be much better than native WAN response times.

Test Procedure

The procedure used to perform the Cache Miss Benchmark ACE Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that WCCPv2 is running on the branch 3 router with the **show ip wccp**, **show ip wccp 61 detail**, and **show ip wccp 62 detail** command.

The output of the **show ip wccp** command should show a value of at least one for the number of cache engines: field for service 61 and 62. If WAE's are redundant, the number of cache engines will reflect the number of WAE's that are configured. From the **show ip wccp** ***service*** detail command verify the WCCP router is Usable.

Step 3 Clear the cache on the each edge WAE.

To do so, open a browser to the WAE GUI, stop the edge service, clear the WAFS cache, and then restart the edge service.

Step 4 Clear the statistics on each of the edge WAEs by issuing the **clear statistics all** command.

Step 5 Launch the WAFS benchmark tool on a Windows client at Branch 3.

Step 6 In the WAFS Benchmark tool, set the use workload files from drive value to Y:.

This is a drive located on a server in DCA.

Step 7 In the WAFS Benchmark tool, set the delay before file save (seconds): value to 15.

Step 8 Click the go button to start the benchmark tool. Each file ranging from 50k to 2MB will be opened from the file server across the WAN. Once opened the application will make a few edits, wait 15 second, save, and then close the file.

Step 9 Verify the WAFS Transport policy is being hit by issuing the **show policy-engine application dynamic** command. Verify in the output you see the TIME_LMT | REPLACE | ACNT_NON_OPT | ACCEPT flags set for the connection to the server on CIFS ports 149 and 445.

Step 10 Save the results, or click open results folder to view the results.

Step 11 Collect statistics on the WAE device(s) at Branch 3 by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE devices to provide transport acceleration for each of the files accessed via CIFS.
- We expect files to open from the data center and not from a locally cached file on the WAE.
- We expect files to open and save considerably faster than they do across a WAN connection with no optimization.

Results

Cache Miss Benchmark ACE Redirection passed.

Redirection

ACE is used as the method of redirection for this project. The ACE in each aggregation switch was configured to send all TCP traffic matching appropriate policies to the WAE's in a round robin fashion.

This section contains the following topics:

- ACE WAAS Configuration and Verification, page 10-8

ACE WAAS Configuration and Verification

Cisco WAAS coupled with Cisco ACE allows large IT organizations to safely consolidate costly infrastructure and improve application performance and scalability for the most demanding high-performance and high-availability environments. The solution provides non disruptive and transparent integration into the networking fabric using existing network devices while helping ensure that value-added network features such as QoS, security, and monitoring features are fully preserved.

The Cisco WAAS and Cisco ACE offers a low total cost of ownership (TCO), high application performance, industry-leading scalability and availability, efficient WAN use, and transparent integration into the network, while also providing secure, centralized management and control in an easy-to-implement package.

This test verified the basic configuration and functionality of ACE redirection to the Core WAE device in Data Center A.

Test Procedure

The procedure used to perform the ACE WAAS Configuration and Verification test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the ACE WAAS rserver and serverfarm configuration on the ACE devices with the **show running-config** command.

Each Core WAE should be defined as an rserver host with the

1. The **transparent** command since NAT should not be used with WAAS. Use the transparent command since NAT should not be enabled for traffic destined to the WAAS devices in the Data Center. ACE should run in dispatch mode with load balancing WAAS devices, using native WAAS IP addresses. Dispatch mode should be enabled by the transparent keyword in the server farm definition.

2. The **predictor hash address source 255.255.255.255** so that traffic flowing from clients to servers is load balanced based on source IP.

3. The **probe** command defined to use ICMP.

Step 3 Verify the configuration on the ACE WAE Vlan for the WAAS serverfarm.

The **ip address**, **alias** IP, and **peer ip address** should all be configured as in any routed mode ACE Vlan configuration.

An ACE security feature, normalization, provides the ability to check for malformed IP and TCP packets. In this setup, ACE is used as a load balancing device. TCP normalization can potentially interfere with WAAS operations, such as sequence number changes and option 21 for auto-discovery. Because of this, **no normalization**, should be configured on the WAE Vlan.

A WAAS device must be in the traffic path, processing traffic bi-directionally. To accomplish this, mac-sticky, which is reverse forwarding path with source MAC address, is configured with the **mac-sticky enable** command.

Finally these service policies should be applied:

- **access-group input ALLOW_Traffic**: To redirect traffic to the WAE
- **servie-policy input L4_WAAS_VIP_POLICY**: To ??
- **service-policy input REMOTE-ACCESS**: To allow ICMP probes

Step 4 Verify the ACE probe configuration is set as the following:

```
probe icmp WAE_ICMP
  interval 2
  faildetect 1
  passdetect interval 2
  passdetect count 1The tuned aggressive timers allow for faster failover in the event of
a failed WAE.
```

Step 5 Verify that the default gateway of the WAEs is the alias IP of the ACE WAE Vlan by issuing the **show running-config** command.

Step 6 Verify each WAE is configured with a standby group interface by issuing the **show interface standby 1** command. The WAE should have its active interface connected to the HSRP active aggregation router and the standby interface connected to the HSRP standby router. Verify by issuing the **show cdp neighbor** command.

Step 7 Verify each WAE has IP/TCP connectivity to the ACE WAE Vlan and the aggregation server Vlan with the **ping** and **telnet** commands.

Step 8 Verify the ACE shows the WAE's OPERATIONAL by issuing the **show serverfarm WAE_FARM** command.

Step 9 Verify the WAE ICMP probe by issuing the **show probe WAE_ICMP detail** command.

Step 10 Verify each WAE is seeing fully optimized connections by issuing the **show tfo connection summary** command. Connections to each branch will be seen if existing connections exits. Connections between data centers will be seen.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect ACE to be configured to redirect to the core WAE.
- We expect the WAEs to be configured with ACE as their default gateway.
- We expect no CPU or memory problems.

Results

ACE WAAS Configuration and Verification passed.

WAFS

Cisco Wide Area File Services (WAFS) software overcomes WAN latency and bandwidth limitations with proprietary Cisco optimization technologies, offering users at branch offices a LAN-like experience when accessing the centralized files over the WAN. This facilitates the consolidation of all branch-office data into central file servers in your data center. WAAS contains a full implementation of the industry-leading WAFS product. The test performed verified the setup of WAFS on the WAAS software platform.

This section contains the following topics:

Cache Hit Benchmark ACE Redirection

The Cisco WAFS Benchmark Tool copies workload files ranging from 50k to 2MB MS Word, Excel, and Powerpoint to the a server and then begins the test. Each file on the server is opened by the client, modified, saved, and closed. Times are recorded for each operation. The cache hit, or warm cache benchmark, tests tests how quickly a file server can be accessed once it has been cached in the local WAFS cache.

The Cache Miss Benchmark test populated the WAE WAFS cache with all the files that will be accessed in this test. In this test the WAFS Benchmark tool was run again and the performance results for a cache hit were verified. WAFS statistics on each WAE were recorded and verified.

Test Procedure

The procedure used to perform the Cache Hit Benchmark ACE Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that WCCPv2 is running on the branch router with the **show ip wccp**, **show ip wccp 61 detail**, and **show ip wccp 62 detail** commands.

The output of the command should show a value of at least one for the number of cache engines: field for service 61 and 62. If WAE's are redundant, the number of cache engines will reflect the number of WAE's that are configured.

Step 3 Clear the statistics on each WAE at the branch by issuing the **clear statistics all** command.

Step 4 Launch the benchmark tool on a Windows client at Branch 3.

Step 5 Set the use workload files from drive value to Z:.

This is the drive used for the testing.

Step 6 Set the delay before file save (seconds): value to 15.

Step 7 Select go to start the benchmark tool. Each file ranging from 50k to 2MB will be opened from the file server across the WAN. Once opened the application will make a few edits, wait 15 second, save, and then close the file.

Step 8 Save the results to file, or click open results folder to view the results.

View the results and verify that the time taken to open, save, and close each file has improved over the Cache Miss Benchmark test.

Step 9 Exit and close the Cisco WAFS Benchmark Tool.

Step 10 Collect statistics on the WAE devices at Branch 3 by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE devices to provide transport acceleration and file services(WAFS) for each of the files accessed via CIFS.
- We expect files to open from the locally cached file on the WAE.
- We expect files to open, save, and close faster than when the WAFS cache is empty.

Results

Cache Hit Benchmark ACE Redirection passed.

CHAPTER 11

Wide Area Application Services (WAAS) WCCP

Cisco Wide Area Application Services 4.0 (WAAS) is a powerful application acceleration and WAN optimization solution for the branch office that improves the performance of any TCP-based application operating in a Wide Area Network (WAN) environment. The WAAS software is built on the WAFS framework and still provides WAFS functionality as well as some added optimization features.

With Cisco WAAS, enterprises can consolidate costly branch office servers and storage into centrally managed data centers, while still offering LAN-like service levels for remote users.

The solution offers a significantly lower total cost of ownership (TCO), greater application performance, more efficient WAN usage, and transparent integration with the network with secure, centralized manageability and control in an easy-to-implement package. Cisco WAAS provides the technologies necessary to enable consolidation of infrastructure into the data center while also providing application acceleration and Wide Area Network (WAN) optimization capabilities that achieve application delivery performance similar to that of a Local Area Network (LAN).

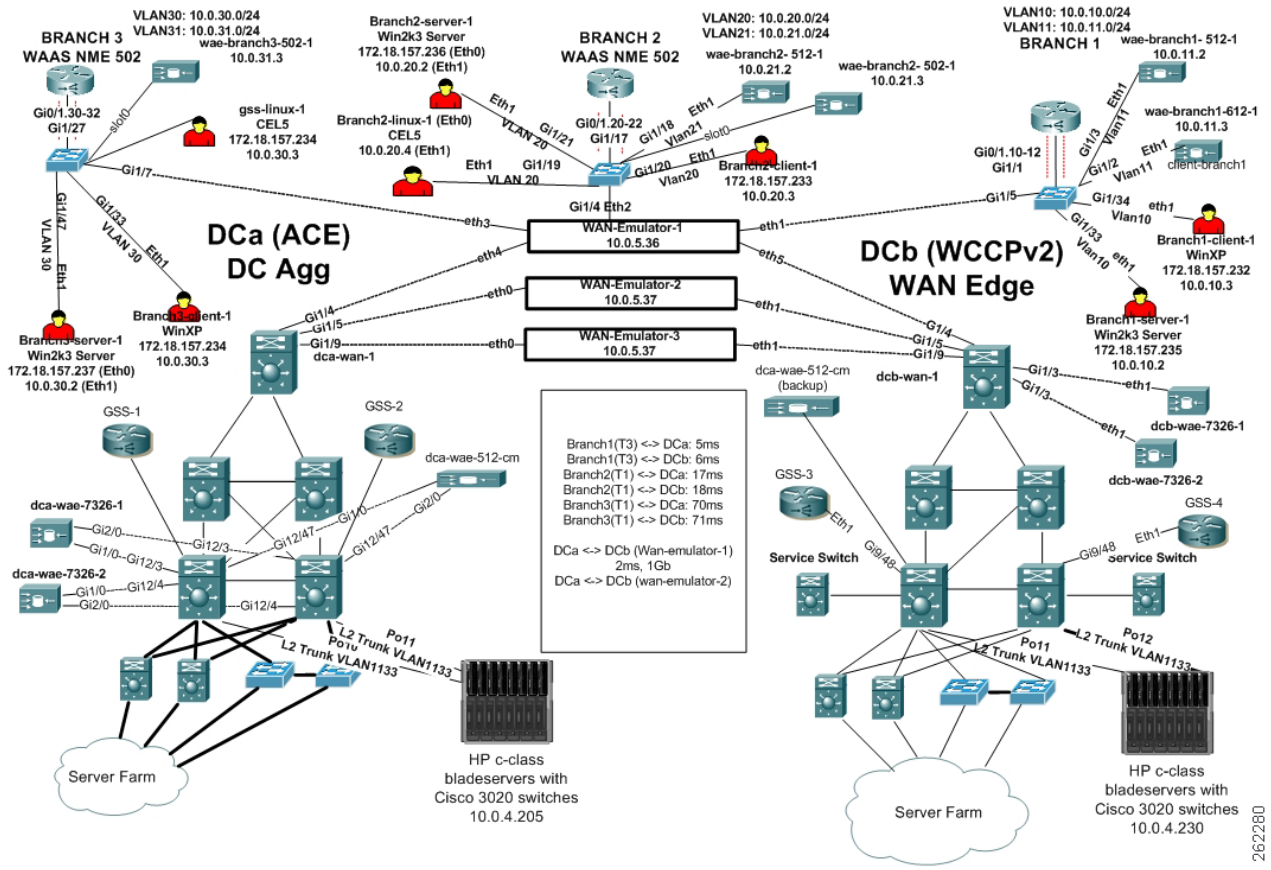
The solution provides the LAN-like performance across the WAN through a combination of technologies, including:

- Application acceleration—Mitigate latency and bandwidth through advanced protocol optimizations, including read-ahead, message prediction, and caching.
- Throughput optimization—Improve behavior of transport protocols to make them more efficient in WAN environments.
- Bandwidth optimization—Minimize the transmission of redundant data patterns through data redundancy elimination (DRE) and compression.

WAAS Topology

Cisco WAAS software running on Cisco Wide Area Application Engine (WAE) platforms is deployed in the data center and remote office locations as appliances attached to the LAN or as network modules (NME-WAE) integrated with the branch router. Cisco WAAS employs the Web Cache Communication Protocol (WCCP) v2, Policy-Based Routing (PBR), or server load balancers suce as the ACE to intercept traffic and transparently forward it to the local Cisco WAE on both sides of the network (Figure 11-1).

Figure 11-1 DCAP WAAS Test Topology



Test Results Summary

Table 11-1 on page 11-3 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 11-1 on page 11-3 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTS's.

A number of resources were referenced during the design and testing phases of Cisco WAAS in DCAP. These include the WAAS Design Guide, produced by Cisco's Enterprise Solution Engineering Data Center team, and Cisco's WAAS Maintenance Guide. Find links to these two documents below.

Enterprise Data Center Wide Area Application Services (WAAS) Design Guide (SRND):

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns377/c649/ccmigration_09186a008081c7da.pdf

Cisco Wide Area Application Services Configuration Guide, Chapter 14: Maintaining Your WAAS System (Maintenance):

http://www.cisco.com/univercd/cc/td/doc/product/webscale/waas/waas407/cfgd/maint.htm

Table 11-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Acceleration, page 11-5	n/a	1. FTP Acceleration Branch 1 2. FTP Acceleration Branch 3 3. HTTP Acceleration Branch 1 4. HTTP Acceleration Branch 3	
Baseline, page 11-10	Device Management, page 11-10	1. SNMP Central Manager MIB Walk WAE512 2. SNMP Core MIB Walk WAE7326 3. SNMP Edge MIB Walk WAE502 4. SNMP Edge MIB Walk WAE512 5. SNMP Edge MIB Walk WAE612	
CIFS, page 11-14	n/a	1. CIFS Cache Hit Benchmark Branch 1 2. CIFS Cache Hit Benchmark Branch 3 3. CIFS Cache Miss Benchmark Branch 1 4. CIFS Cache Miss Benchmark Branch 3 5. CIFS Native WAN Benchmark Branch 1 6. CIFS Native WAN Benchmark Branch 3 7. CIFS Verification WAE502 8. CIFS Verification WAE612	
NTP, page 11-25	n/a	1. NTP Configuration and Functionality	

Table 11-1 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Reliability, page 11-27	n/a	1. Central Manager Reload WAE512 2. Core Reload WAE7326 3. Edge Reload WAE502 4. Edge Reload WAE512	CSCsi69388 CSCsi75538,
Upgrade, page 11-30	n/a	1. Core CLI Upgrade WAE612 2. Edge GUI Upgrade WAE512	
WAFS, page 11-33	n/a	1. WAFS Configuration Verification	
WCCPv2, page 11-35	n/a	1. WCCPv2 Basic Configuration on Edge WAE2821 2. WCCPv2 Configuration and Functionality on Core Sup720 3. WCCPv2 Configuration and Functionality on Core WAE7326 4. WCCPv2 Configuration and Functionality on Edge WAE 512 5. WCCPv2 Configuration and Functionality on Edge WAE3845	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Acceleration

Cisco Wide Area Application Services 4.0 (WAAS) is a powerful application acceleration and WAN optimization solution for the branch office that improves the performance of any TCP-based application operating in a Wide Area Network (WAN) environment. The WAAS software is built on the WAFS framework and still provides WAFS functionality as well as some added optimization features. With WAAS WAN traffic is optimized in three different ways, TCP Flow Optimization (TFO), Data Redundancy Elimination (DRE), and LZ Compression.

This section contains the following topics:

FTP Acceleration Branch 1

This test verified the ability of WAAS to accelerate FTP traffic. From Branch 1, 10 simulated clients initiating an FTP session to a server in the data center, issuing the bin, pwd, cd, ls, and finally get commands. The client makes a request for a 1MB file. The traffic was run continuously for 3 minutes with and without optimization.

It was verified and quantified that FTP response times were improved when WAAS acceleration was enabled compared to native WAN response times. Branch 1 had 8ms latency and T3 bandwidth to the file server in the data center.

Test Procedure

The procedure used to perform the FTP Acceleration Branch 1 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the branch router disable WCCP redirects with the **no ip wccp 61** and **no ip wccp 62** commands. This will cause all traffic to be unoptimized by WAAS.

Step 3 With the Test Test tool begin a traffic stream. The stream should be set up to simulate 10 users each making 1 FTP GET request for a 1MB file continuously for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 4 Save the results for the entire group as well as one individual client. The group statistics will show the overall performance of Input and Output Bits/s. The individual client statistics will show more the same I/O statistics for that individual client and will also include the Mean Get Time for each FTP transaction.

Step 5 On the branch router enable WCCP redirection with the **ip wccp 61** and **ip wccp 62** commands. Verify WCCP State becomes usable with the **show ip wccp 61 detail** and **show ip wccp 62 detail** commands.

Step 6 Clear the statistics on the WAE at the branch by issuing the **clear statistics all** command.

Step 7 With the Shenick Test tool begin a traffic stream. The stream should be set up to simulate 10 users each making 1 FTP GET request for a file in the 500K-1MB range every 10ms for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 8 While the traffic is running verify WAE that the connections are being optimized by issuing the **show tfo connection summary** command.

Step 9 Save the results for the entire group as well as one individual client. The group statistics will show the overall performance of Input and Output Bits/s. The individual client statistics will show more the same I/O statistics for that individual client and will also include the Mean Get Time for each FTP transaction.

Step 10 Collect statistics on the WAE devices at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dreVerify that TFO savings were seen for inbound and outbound Web based
traffic. Also verify that DRE savings were measured appropriately.
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that FTP traffic will pass when WAAS is enabled.
- We expect that FTP GET response times will decrease with WAAS acceleration enabled.

Results

FTP Acceleration Branch 1 passed.

FTP Acceleration Branch 3

This test verified the ability of WAAS to accelerate FTP traffic. From Branch 3, 10 simulated clients initiating an FTP session to a server in the data center, issuing the bin, pwd, cd, ls, and finally get commands. The client makes a request for a randomly sized file between 500K and 1MB. There is a 5ms delay between commands and a 10ms delay between sessions. The traffic was run for 3 minutes with and without optimization.

It was verified and quantified that FTP response times were improved when WAAS acceleration was enabled compared to native WAN response times. Branch 2 had 70ms latency and T1 bandwidth to the file server in the data center.

Test Procedure

The procedure used to perform the FTP Acceleration Branch 3 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the branch router disable WCCP redirects with the **no ip wccp 61** and **no ip wccp 62** commands. This will cause all traffic to be unoptimized by WAAS.

Step 3 With the Test Test tool begin a traffic stream. The stream should be set up to simulate 10 users each making 1 FTP GET request for a 1MB file continually for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 4 Save the results for the entire group as well as one individual client. The group statistics will show the overall performance of Input and Output Bits/s. The individual client statistics will show more the same I/O statistics for that individual client and will also include the Mean Get Time for each FTP transaction.

Step 5 On the branch router enable WCCP redirection with the **ip wccp 61** and **ip wccp 62** commands. Verify WCCP State becomes usable with the **show ip wccp 61 detail** and **show ip wccp 62 detail** commands.

Step 6 Clear the statistics on the WAE at the branch by issuing the **clear statistics all** command.

Step 7 With the Shenick Test tool begin a traffic stream. The stream should be set up to simulate 10 users each making 1 FTP GET request for a 1MB file continually for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 8 While the traffic is running verify WAE that the connections are being optimized by issuing the **show tfo connection summary** command.

Step 9 Save the results for the entire group as well as one individual client. The group statistics will show the overall performance of Input and Output Bits/s. The individual client statistics will show more the same I/O statistics for that individual client and will also include the Mean Get Time for each FTP transaction.

Step 10 Collect statistics on the WAE devices at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dreVerify that TFO savings were seen for inbound and outbound Web based
traffic. Also verify that DRE savings were measured appropriately.
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that FTP traffic will pass when WAAS is enabled.
- We expect that FTP GET response times will decrease with WAAS acceleration enabled.

Results

FTP Acceleration Branch 3 passed.

HTTP Acceleration Branch 1

This test verified the ability of WAAS to accelerate HTTP traffic. From Branch 1, 20 simulated clients making requests of 64k HTTP GET connections were started. The connection from each client was established, a single HTTP GET [Request] was made, and then the connection was torn down by the client. The traffic was run for 3 minutes with and without optimization.

It was verified and quantified that HTTP response times were improved when WAAS acceleration was enabled compared to native WAN response times. Branch 1 had 6ms latency and T3 bandwidth to the file server in the data center.

Test Procedure

The procedure used to perform the HTTP Acceleration Branch 1 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the branch router disable WCCP redirects with the **no ip wccp 61** and **no ip wccp 62** commands. This will cause all traffic to be unoptimized by WAAS.

Step 3 With the Test tool begin a traffic stream. The stream should be set up to simulate 10 users each making 1 HTTP GET request for a file of 64k in size continually for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 4 Save the results for the entire group as well as one individual client. The group statistics will show the overall performance of Input and Output Bits/s. The individual client statistics will show more the same I/O statistics for that individual client and will also include the Mean Get Time for each HTTP transaction.

Step 5 On the branch router enable WCCP redirection with the **ip wccp 61** and **ip wccp 62** commands. Verify WCCP State becomes usable with the **show ip wccp 61 detail** and **show ip wccp 62 detail** commands.

Step 6 Clear the statistics on the WAE at the branch by issuing the **clear statistics all** command.

Step 7 With the Test tool begin a traffic stream. The stream should be set up to simulate 10 users each making 1 HTTP GET request for a file of 64k in size continually for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 8 While the traffic is running verify WAE that the connections are being optimized by issuing the **show tfo connection summary** command.

Step 9 Save the results for the entire group as well as one individual client. The group statistics will show the overall performance of Input and Output Bits/s. The individual client statistics will show more the same I/O statistics for that individual client and will also include the Mean Get Time for each HTTP transaction.

Step 10 Collect statistics on the WAE devices at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dreVerify that TFO savings were seen for inbound and outbound Web based
traffic. Also verify that DRE savings were measured appropriately.
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that HTTP traffic will pass when WAAS is enabled.
- We expect that HTTP GET response times will decrease with WAAS acceleration enabled.

Results

HTTP Acceleration Branch 1 passed.

HTTP Acceleration Branch 3

This test verified the ability of WAAS to accelerate HTTP traffic. From Branch 3, 10 simulated clients making requests of 64k HTTP GET connections were started. The connection from each client was established, a single HTTP GET [Request] was made, and then the connection was torn down by the client. The traffic was run for 3 minutes with and without optimization.

It was verified and quantified that HTTP response times were improved when WAAS acceleration was enabled compared to native WAN response times.

Test Procedure

The procedure used to perform the HTTP Acceleration Branch 3 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the branch router disable WCCP redirects with the **no ip wccp 61** and **no ip wccp 62** commands. This will cause all traffic to be unoptimized by WAAS.

Step 3 With the Test tool begin a traffic stream. The stream should be set up to simulate 20 users each making 1 HTTP GET request for a file of 64k continually for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 4 Save the results for the entire group as well as one individual client. The group statistics will show the overall performance of Input and Output Bits/s. The individual client statistics will show more the same I/O statistics for that individual client and will also include the Mean Get Time for each HTTP transaction.

Step 5 On the branch router enable WCCP redirection with the **ip wccp 61** and **ip wccp 62** commands. Verify WCCP State becomes usable with the **show ip wccp 61 detail** and **show ip wccp 62 detail** commands.

Step 6 Clear the statistics on the WAE by issuing the **clear statistics all** command.

Step 7 With the Test tool begin a traffic stream. The stream should be set up to simulate 20 users each making 1 HTTP GET request for a file of 64k continually for a duration of 3 minutes. Each TCP connection is established and torn down gracefully after the GET has been attempted.

Step 8 While the traffic is running verify WAE that the connections are being optimized by issuing the **show tfo connection summary** command.

Step 9 Save the results for the entire group as well as one individual client. The group statistics will show the overall performance of Input and Output Bits/s. The individual client statistics will show more the same I/O statistics for that individual client and will also include the Mean Get Time for each HTTP transaction.

Step 10 Collect statistics on the WAE devices at Branch 3 by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dreVerify that TFO savings were seen for inbound and outbound Web based
traffic. Also verify that DRE savings were measured appropriately.
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that HTTP traffic will pass when WAAS is enabled.
- We expect that HTTP GET response times will decrease with WAAS acceleration enabled.

Results

HTTP Acceleration Branch 3 passed.

Baseline

Baseline tests verify network is in working order prior to starting testing and quantify steady state network performance.

This section contains the following topics:

Device Management

Simple Network Management Protocol (SNMP) is an application-layer protocol designed to facilitate the exchange of management information between network devices.

This section contains the following topics:

SNMP Central Manager MIB Walk WAE512

Simple Network Management Protocol (SNMP) is ubiquitous as a tool used by network administrators to manage network performance, find and solve network problems, and plan for network growth.

This test verified that 1000 SNMP walk'a of the MIB tree of a WAE-512 device did not cause any tracebacks or crashes. From a server, 1000 version 1 SNMP walks were performed on the device.

Test Procedure

The procedure used to perform the SNMP Central Manager MIB Walk WAE512 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the SNMP configuration of dca-wae-512-cm using the **show running-config** command.

Step 3 From the lickskillet server CLI perform 1000 SNMP walks on the DUT using the **snmpwalk** utility.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all SNMP walks will run without error.
- We expect that no tracebacks or crashes will occur on the DUT.

Results

SNMP Central Manager MIB Walk WAE512 passed.

SNMP Core MIB Walk WAE7326

SNMP is ubiquitous as a tool used by network administrators to manage network performance, find and solve network problems, and plan for network growth.

This test verified that 1000 SNMP walks of the MIB tree of a WAE-7326 device did not cause any tracebacks or reloads. From a server, 1000 version 1 SNMP walks were performed on the device.

Test Procedure

The procedure used to perform the SNMP Core MIB Walk WAE7326 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the SNMP configuration of dca-wae-7326-1 using the **show running-config** command.

Step 3 From the lickskillet server CLI perform 1000 SNMP walks on the DUT using the SNMPwalks utility.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all SNMP walks to run without error.
- We expect no tracebacks or reloads to occur on the DUT.

Results

SNMP Core MIB Walk WAE7326 passed.

SNMP Edge MIB Walk WAE502

Simple Network Management Protocol (SNMP) is ubiquitous as a tool used by network administrators to manage network performance, find and solve network problems, and plan for network growth.

This test verified that 1000 SNMP walk'a of the MIB tree of a WAE-502 device did not cause any tracebacks, or crashes. From a server, 1000 version 1 SNMP walks were performed on the device.

Test Procedure

The procedure used to perform the SNMP Edge MIB Walk WAE502 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the SNMP configuration of wae-branch2-502-1 using the **show running-config** command.

Step 3 From the lickskillet server CLI perform 1000 SNMP walks on the DUT using the **snmpwalk** utility.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all SNMP walks will run without error.
- We expect that no tracebacks or crashes will occur on the DUT.

Results

SNMP Edge MIB Walk WAE502 passed.

SNMP Edge MIB Walk WAE512

SNMP is ubiquitous as a tool used by network administrators to manage network performance, find and solve network problems, and plan for network growth.

This test verified that 1000 SNMP walks of the MIB tree of a WAE-512 device did not cause any memory tracebacks or reloads. From a server, 1000 version 1 SNMP walks were performed on the device.

Test Procedure

The procedure used to perform the SNMP Edge MIB Walk WAE512 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the SNMP configuration of wae-branch1-512-1 using the **show running-config** command.

Step 3 From the lickskillet server CLI perform 1000 SNMP walks on the DUT using the SNMPwalk utility.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all SNMP walks will run without error.
- We expect no tracebacks or reloads to occur on the DUT.

Results

SNMP Edge MIB Walk WAE512 passed.

SNMP Edge MIB Walk WAE612

SNMP is ubiquitous as a tool used by network administrators to manage network performance, find and solve network problems, and plan for network growth.

This test verified that 1000 SNMP walks of the MIB tree of a WAE-612 device did not cause any tracebacks or reloads. From a server, 1000 version 1 SNMP walks were performed on the device.

Test Procedure

The procedure used to perform the SNMP Edge MIB Walk WAE612 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the SNMP configuration of wae-branch1-612-1 using the **show running-config** command.

Step 3 From the lickskillet server CLI perform 1000 SNMP walks on the DUT using the SNMPwalk utility.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all SNMP walks to run without error.
- We expect no tracebacks or reloads to occur on the DUT.

Results

SNMP Edge MIB Walk WAE612 passed.

CIFS

Common Internet File System (CIFS) is a protocol that defines a standard for remote file access. With CIFS, users with different platforms and computers can share files without having to install new software. CIFS runs over TCP/IP but uses the SMB (Server Message Block) protocol found in Microsoft Windows for file and printer access; therefore, CIFS will allow all applications, not just Web browsers, to open and share files across the Internet. With CIFS, changes made to a file are simultaneously saved on both the client and server side. Clients in a WAAS network use the CIFS cache service to request file and print services from servers over a network. The tests performed verified the functionality and baseline performance for CIFS with, and without, the WAAS software.

This section contains the following topics:

CIFS Cache Hit Benchmark Branch 1

The Cisco WAFS Benchmark Tool copies workload files ranging from 50k to 2MB MS Word, Excel, and Powerpoint to the file server, and begins the test. Each file is opened, modified, saved, and closed, and times are recorded for each operation.

The WAAS Cache Hit, which is also known as warm cache, benchmark, tests how quickly a file server can be subsequently accessed through the WAFS cache (data has already been accesse once and is in the local cache).

The Cache Miss Benchmark test populated the WAE cache with all the files that were accessed in this test. The WAFS Benchmark tool was then run and the performance results for a cache hit were verified.

Branch 1 was simulated to have T3 bandwidth and 8ms of latency to the file server in the data center.

Test Procedure

The procedure used to perform the CIFS Cache Hit Benchmark Branch 1 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that WCCPv2 is running on wae-3845-branch1 and dcb-wan-1 with the show ip wccp command.

The output of the command should show a value of at least one for the number of cache engines: field for service 61 and 62. If WAE's are redundant, the number of cache engines will reflect the number of WAE's that are configured.

Step 3 Clear the statistics on the WAE's at Branch 1 by issuing the **clear statistics all** command.

Step 4 Launch the benchmark tool on a Windows client at Branch 1.

Step 5 Set the use workload files from drive value to X:.

This is the drive used for the testing.

Step 6 Set the delay before file save (seconds): value to 15.

Step 7 Select go to start the benchmark tool. Each file ranging from 50k to 2MB will be opened from the file server across the WAN. Once opened the application will make a few edits, wait 15 second, save, and then close the file.

Step 8 Save the results to file, or click open results folder to view the results.

View the results and verify that the time taken to open, save, and close each file has improved over the Cache Miss Benchmark test.

Step 9 Exit and close the Cisco WAFS Benchmark Tool.

Step 10 Collect statistics on the WAE devices at Branch 1 by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE devices to provide transport acceleration and file services(WAFS) for each of the files accessed via CIFS.
- We expect files to open from the locally cached file on the WAE.
- We expect files to open, save, and close faster than when the WAFS cache is empty.

Results

CIFS Cache Hit Benchmark Branch 1 passed.

CIFS Cache Hit Benchmark Branch 3

The Cisco WAFS Benchmark Tool copies workload files ranging from 50k to 2MB MS Word, Excel, and Powerpoint to the file server, and begins the test. Each file is opened, modified, saved, and closed, and times are recorded for each operation.

The WAAS Cache Hit, which is also known as warm cache, benchmark, tests how quickly a file server can be subsequently accessed through the WAFS cache (data has already been accesse once and is in the local cache).

The Cache Miss Benchmark test populated the WAE cache with all the files that were accessed in this test. The WAFS Benchmark tool was then run and the performance results for a cache hit were verified.

Branch 3 was simulated to have T1 bandwidth and 70ms of latency to the file server in the Data Center.

Test Procedure

The procedure used to perform the CIFS Cache Hit Benchmark Branch 3 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that WCCPv2 is running on wae-2811-branch3 and dcb-wan-1 with the **show ip wccp**, **show ip wccp 61 detail**, and **show ip wccp 62 detail** commands.

The output of the command should show a value of at least one for the number of cache engines: field for service 61 and 62. If WAE's are redundant, the number of cache engines will reflect the number of WAE's that are configured.

Step 3 Clear the statistics on the WAE's at Branch 3 by issuing the **clear statistics all** command.

Step 4 Launch the benchmark tool on a Windows client at Branch 3.

Step 5 Set the use workload files from drive value to X:.

This is the drive used for the testing.

Step 6 Set the delay before file save (seconds): value to 15.

Step 7 Select go to start the benchmark tool. Each file ranging from 50k to 2MB will be opened from the file server across the WAN. Once opened the application will make a few edits, wait 15 second, save, and then close the file.

Step 8 Save the results to file, or click open results folder to view the results.

View the results and verify that the time taken to open, save, and close each file has improved over the Cache Miss Benchmark test.

Step 9 Exit and close the Cisco WAFS Benchmark Tool.

Step 10 Collect statistics on the WAE devices at Branch 3 by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE devices to provide transport acceleration and file services(WAFS) for each of the files accessed via CIFS.
- We expect files to open from the locally cached file on the WAE.
- We expect files to open, save, and close faster than when the WAFS cache is empty.

Results

CIFS Cache Hit Benchmark Branch 3 passed.

CIFS Cache Miss Benchmark Branch 1

The Cisco WAFS Benchmark Tool copies workload files ranging from 50k to 2MB MS Word, Excel, and Powerpoint to the file server, and begins the test. Each file is opened, modified, saved, and closed, and times are recorded for each operation.

The Cache Miss, or cold cache, Benchmark Tests tests how quickly a file server can be accessed through the WAAS cache for the first time (with no data in the cache).

In this test the WAFS Benchmark tool was ran while WCCPv2 redirection to the WAE device in the CORE and EDGE was enabled. The WAE edge cache first cleared, and then the Benchmark tool was started. This provided performance results for files that were not cached yet the TCP flow was optimized.

Branch 1 was simulated to have T3 bandwidth and 8ms of latency to the file server in the data center.

Test Procedure

The procedure used to perform the CIFS Cache Miss Benchmark Branch 1 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that WCCPv2 is running on wae-3845-branch1 and dca-wan-1 with the **show ip wccp** command.

The output of the command should show a value of at least one for the number of cache engines: field for service 61 and 62. If WAE's are redundant, the number of cache engines will reflect the number of WAE's that are configured.

Step 3 Clear the cache on the edge WAE's.

Navigate to the WAE GUI, stop the edge service, clear the WAFS cache, and then restart the edge service.

Step 4 Clear the statistics on the WAE's at Branch 1 by issuing the **clear statistics all** command.

Step 5 Launch the benchmark tool on a Windows client at Branch 1.

Step 6 Set the use workload files from drive value to X:.

This is the drive used for the testing.

Step 7 Set the delay before file save (seconds): value to 15.

Step 8 Select go to start the benchmark tool. Each file ranging from 50k to 2MB will be opened from the file server across the WAN. Once opened the application will make a few edits, wait 15 second, save, and then close the file.

Step 9 Save the results, or click open results folder to view the results.

Step 10 Collect statistics on the WAE devices at Branch 1 by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE devices to provide optimization.
- We expect files to open and save considerably faster than they do across a WAN connection with no optimization.

Results

CIFS Cache Miss Benchmark Branch 1 passed.

CIFS Cache Miss Benchmark Branch 3

The Cisco WAFS Benchmark Tool copies workload files ranging from 50k to 2MB MS Word, Excel, and Powerpoint to the file server, and begins the test. Each file is opened, modified, saved, and closed. Times are recorded for each operation.

The Cache Miss, or cold cache, Benchmark Tests tests how quickly a file server can be accessed for the first time (with no data in the cache).

In this test, the WAFS Benchmark tool was run while WCCPv2 redirection to the WAE device in the CORE and EDGE was enabled. The WAE edge cache was first cleared, and then the Benchmark tool was started. This provided performance results for files that were not cached yet the TCP flow was optimized.

Branch 3 was simulated to have T1 bandwidth and 70ms of latency to the file server in the data center.

Test Procedure

The procedure used to perform the CIFS Cache Miss Benchmark Branch 3 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that WCCPv2 is running on wae-2811-branch3 and dca-wan-1 with the **show ip wccp**, **show ip wccp 61 detail**, and **show ip wccp 62 detail** command.

The output of the **show ip wccp**command should show a value of at least one for the number of cache engines: field for service 61 and 62. If WAE's are redundant, the number of cache engines will reflect the number of WAE's that are configured. From the **show ip wccp *service*** detail command verify the WCCP router is Usable.

Step 3 Clear the cache on the edge WAE's.

Navigate to the WAE GUI, stop the edge service, clear the WAFS cache, and then restart the edge service.

Step 4 Clear the statistics on the WAE's at Branch 3 by issuing the **clear statistics all** command.

Step 5 Launch the WAFS benchmark tool on a Windows client at Branch 3.

Step 6 In the WAFS Benchmark tool, set the use workload files from drive value to X:.

This is the drive used for the testing.

Step 7 In the WAFS Benchmark tool, set the delay before file save (seconds): value to 15.

Step 8 Click the go button to start the benchmark tool. Each file ranging from 50k to 2MB will be opened from the file server across the WAN. Once opened the application will make a few edits, wait 15 second, save, and then close the file.

Step 9 Save the results, or click open results folder to view the results.

Step 10 Collect statistics on the WAE devices at Branch 3 by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE devices to provide transport acceleration for each of the files accessed via CIFS.
- We expect files to open from the data center and not from a locally cached file on the WAE.
- We expect files to open and save considerably faster than they do across a WAN connection with no optimization.

Results

CIFS Cache Miss Benchmark Branch 3 passed.

CIFS Native WAN Benchmark Branch 1

The Cisco WAFS Benchmark Tool copies workload files ranging from 50k to 2MB MS Word, Excel, and Powerpointto the file server, and begins the test. Each file is opened, modified, saved, and closed, and times are recorded for each operation.

This test verified how quickly a file server can be accessed directly over the WAN. For this test, WCCP was not configured on the core and branch routers so that no acceleration took place. The results express the performance what would be seen normally over a T3 connection with 7ms RTT latency.

Test Procedure

The procedure used to perform the CIFS Native WAN Benchmark Branch 1 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Unconfigure WCCPv2 on wae-3845-branch1 and dca-wan-1 with the **no ip wccp 61** and **no ip wccp 62** commands.

This will cause all WAN traffic to not hit the WAE devices.

Step 3 From the client at branch1 verify the latency and bandwidth to the file server by using the **ping** and **ftp** commands.

Step 4 Launch the benchmark tool on a Windows client at Branch 1.

Step 5 Check the prepare benchmark box.

Step 6 Set the copy workload files to drive value to X:\.

This is the target directory on the file server where the files are copied so that tests can be run on them.

Step 7 Uncheck Prepare benchmark and check the run benchmark box.

Step 8 Set the use workload files from drive value to X:.

This is the drive used for the testing.

Step 9 Set the delay before file save (seconds): value to 15.

Step 10 Select go to start the benchmark tool. Each file ranging from 50k to 2MB will be opened from the file server across the WAN. Once opened the application will make a few edits, wait 15 second, save, and then close the file.

Step 11 Click save results to file to save the results, or click open results folder to view the results.

View the results.

Step 12 Exit and close the Cisco WAFS Benchmark Tool.

Step 13 Reconfigure WCCPv2 on wae-3845-branch1 and dca-wan-1 with the **ip wccp 61** and **ip wccp 62** commands.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect opened files to take considerably longer than when WAFS/CIFS acceleration is configured.

Results

CIFS Native WAN Benchmark Branch 1 passed.

CIFS Native WAN Benchmark Branch 3

The Cisco WAFS Benchmark Tool copies workload files ranging from 50k to 2MB MS Word, Excel, and Powerpointto the file server, and begins the test. Each file is opened, modified, saved, and closed, and times are recorded for each operation.

This test verified how quickly a file server can be accessed directly over the WAN. For this test, WCCP was not configured on the core and branch routers so that no acceleration took place. The results express the performance what would be seen normally over a T1 connection with 70ms RTT latency.

Test Procedure

The procedure used to perform the CIFS Native WAN Benchmark Branch 3 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Unconfigure WCCPv2 on wae-2811-branch3 and dca-wan-1 with the **no ip wccp 61** and **no ip wccp 62** commands.

This will cause all WAN traffic to be unoptimized.

Step 3 From the client at Branch 3 verify the latency and bandwidth to the file server by using the **ping** and **ftp** commands.

Step 4 Launch the benchmark tool on a Windows client at Branch 3.

Step 5 Check the prepare benchmark box.

Step 6 Set the copy workload files to drive value to X:\.

This is the target directory on the file server where the files are copied so that tests can be run on them.

Step 7 Uncheck Prepare benchmark and check the run benchmark box.

Step 8 Set the use workload files from drive value to X:.

This is the drive used for the testing.

Step 9 Set the delay before file save (seconds): value to 15.

Step 10 Select go to start the benchmark tool. Each file ranging from 50k to 2MB will be opened from the file server across the WAN. Once opened the application will make a few edits, wait 15 second, save, and then close the file.

Step 11 Click save results to file to save the results, or click open results folder to view the results.

View the results.

Step 12 Exit and close the Cisco WAFS Benchmark Tool.

Step 13 Reconfigure WCCPv2 on wae-2811-branch1 and dca-wan-1 with the **ip wccp 61** and **ip wccp 62** commands.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect open, save, and close times of all files to take considerably longer than when WAFS/CIFS acceleration is configured.

Results

CIFS Native WAN Benchmark Branch 3 passed.

CIFS Verification WAE502

CIFS enables collaboration on the Internet by defining a remote file access protocol that is compatible with the way applications already share data on local disks and network file servers. CIFS incorporates the same high-performance, multiuser read and write operations, locking, and file-sharing semantics that are the backbone of today's sophisticated enterprise computer networks. CIFS runs over TCP/IP and utilizes the internet's global Domain Naming Service (DNS) for scalability, and is optimized to support slower speed dial up connections common on the internet.

WAAS has an embedded flow protection mechanism to ensure that existing CIFS sessions will not be broken when the device is brought online or additional WAE devices join the WCCP service groups.

CIFS sessions that were not established while the WAE's were fully online and accelerating will not be CIFS accelerated and the redirected CIFS traffic will be returned to the router for native processing (DRE/TFO/LZ may be applied, assuming the CIFS-non-wafs policy is configured accordingly). To ensure CIFS sessions are fully accelerated, the CIFS session needs to be established after the WAE's are online, optimizing, and configured to accelerate CIFS. If the connection was established before the WAE came online, this connection will not be accelerated, it will be a passed-through connection ("In Progress").

This test verified that CIFS acceleration was working for a windows client located at remote branch 3 connected to the data center by a emulated T1 connection with approximately 70ms of RTT latency. A WAE502 was installed at the branch LAN, configured for CIFS acceleration and WCCP redirects were turned on. It was verified that the WAE device accessed the file server in the data center using CIFS acceleration. The CIFS connections established on the file server were verified that they came from the Core WAE and not the remote client. CIFS auto-discovery was verified for the established connection.

Test Procedure

The procedure used to perform the CIFS Verification WAE502 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 In the central-manager verify that the WAE is configured as an edge device with connectivity to a core cluster comprised of the Core DC WAE's. Navigate in the GUI as follows:

Services -> File -> Connectivy

Verify that the Edge WAE is assigned as a member and is online. To verify its online status click on the core cluster name and then Assign Edge Devices. The WAE should have a check next to it and the Status should be Online. If necessary add the edge device.

Step 3 On the Edge WAE verify that connectivity to the Core Cluster is established. Open up a browser to the Edge WAE and click the Monitoring tab on the sidebar. Verify the Core Cluster exists and that under the Connected column a green check mark appears.

Step 4 On the branch client at branch 3 open up a file being shared on a file server located in DCa.

NOTE: Make sure digital signatures are disabled on the file server. CIFS auto-discover will fail if these signatures are enabled.

Step 5 Verify on the Edge WAE, that in the output of the **show policy-engine application dynamic** there are entries for the server you trying to connect to and there is a Flag with value ACCEPT.

Note: The record for a file server remains in the dynamic map for three minutes after the last connection to it is closed.

Step 6 On the file server Use Microsoft Management Console to inspect the name or IP of the computer that opened the CIFS session. If you see the IP address of the Core WAE, it means that the CIFS session is being accelerated by WAAS.

If the IP address of the Windows client appears under the computer section, then it means that the session is connected directly without acceleration. The session would need to be reestablished so acceleration can be applied.

Step 7 Inspect the CIFS statistics on the WAE device GUI. Statistics should be incrementing when accessing files or folders (number of open files/sessions/remote/local request should increment).

To check the statistics open browser to the Edge WAE and navigate as follows: WAFS Edge -> Monitoring -> CIFS

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect auto-discovery on the edge WAE to have identified the connection to the server on ports 139 and 445.
- We expect on the domain controller sessions to the core WAE to be established.
- We expect on the domain controller no sessions to the client to be established.
- We expect the statistics on the edge WAE to show an increase in CIFS accesses once files have been transferred.

Results

CIFS Verification WAE502 passed.

CIFS Verification WAE612

CIFS enables collaboration on the Internet by defining a remote file access protocol that is compatible with the way applications already share data on local disks and network file servers. CIFS incorporates the same high-performance, multiuser read and write operations, locking, and file-sharing semantics that are the backbone of today's sophisticated enterprise computer networks. CIFS runs over TCP/IP and utilizes the internet's global Domain Naming Service (DNS) for scalability, and is optimized to support slower speed dial up connections common on the internet.

WAAS has an embedded flow protection mechanism to ensure that existing CIFS sessions will not be broken when the device is brought online or additional WAE devices join the WCCP service groups.

CIFS sessions that were not established while the WAE's were fully online and accelerating will not be CIFS accelerated and the redirected CIFS traffic will be returned to the router for native processing (DRE/TFO/LZ may be applied, assuming the CIFS-non-wafs policy is configured accordingly). To ensure CIFS sessions are fully accelerated, the CIFS session needs to be established after the WAE's are online, optimizing, and configured to accelerate CIFS. If the connection was established before the WAE came online, this connection will not be accelerated, it will be a passed-through connection ("In Progress").

This test verified that CIFS acceleration was working for a windows client located at remote branch 2 connected to the data center by a emulated T3 connection with approximately 7ms of RTT latency. A WAE512's was installed at the branch LAN, configured for CIFS acceleration and WCCP redirects were turned on. It was verified that the WAE device accessed the file server in the data center using CIFS acceleration. The CIFS connections established on the file server were verified that they came from the Core WAE and not the remote client. CIFS auto-discovery was verified for the established connection.

Test Procedure

The procedure used to perform the CIFS Verification WAE612 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 In the central-manager verify that the WAE is configured as an edge device with connectivity to a core cluster comprised of the Core DC WAE's. Navigate in the GUI as follows: Services -> File -> Connectivy Verify that the Edge WAE is assigned as a member and is online. To verify its online status click on the core cluster name and then Assign Edge Devices. The WAE should have a check next to it and the Status should be Online. If necessary add the edge device.

Step 3 On the Edge WAE verify that connectivity to the Core Cluster is established. Open up a browser to the Edge WAE and click the Monitoring tab on the sidebar. Verify the Core Cluster exists and that under the Connected column a green check mark appears.

Step 4 On the branch client at branch 1 open up a file being shared on a file server located in DCa.

NOTE: Make sure digital signatures are disabled on the file server. CIFS auto-discover will fail if these signatures are enabled.

Step 5 Verify on the Edge WAE, that in the output of the **show policy-engine application dynamic** there are entries for the server you trying to connect to and there is a Flag with value ACCEPT.

Note: The record for a file server remains in the dynamic map for three minutes after the last connection to it is closed.

Step 6 On the file server Use Microsoft Management Console to inspect the name or IP of the computer that opened the CIFS session. If you see the IP address of the Core WAE, it means that the CIFS session is being accelerated by WAAS.

If the IP address of the Windows client appears under the computer section, then it means that the session is connected directly without acceleration. The session would need to be reestablished so acceleration can be applied.

Step 7 Inspect the CIFS statistics on the WAE device GUI. Statistics should be incrementing when accessing files or folders (number of open files/sessions/remote/local request should increment).

To check the statistics open browser to the Edge WAE and navigate as follows: WAFS Edge -> Monitoring -> CIFS

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect auto-discovery on the edge WAE to have identified the connection to the server on ports 139 and 445.
- We expect on the domain controller sessions to the core WAE to be established.
- We expect on the domain controller no sessions to the client to be established.
- We expect the statistics on the edge WAE to show an increase in CIFS accesses once files have been transferred.

Results

CIFS Verification WAE612 passed.

NTP

Network Time Protocol (NTP) synchronizes timekeeping among a set of distributed time servers and clients. This synchronization allows events to be correlated when system logs are created and other time-specific events occur. An NTP server must be accessible by the client switch. NTP runs over User Datagram Protocol (UDP), which runs over IP.

This section contains the following topics:

NTP Configuration and Functionality

NTP synchronizes timekeeping among a set of distributed time servers and clients. This synchronization allows events to be correlated when system logs are created and other time specific events occur. In order for WAFS/WAAS to work correctly all the devices clocks must be synchronized using NTP. Because time stamps play an important role in determining whether or not the data being accessed has been changed since the last time it was accessed.

This test verified the basic configuration and functionality of NTP on the central manager, core, and edge devices. Through the central manager GUI, the central manager was first synchronized to a NTP server that was used throughout the lab networks. Each WAE device was then configured through the GUI to be synchronized to the central manager. The configuration and clock for each device was verified via the CLI.

Test Procedure

The procedure used to perform the NTP Configuration and Functionality test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Navigate to the Windows Name Services page in the CM GUI as follows:

Devices -> select the Central Manager(CM) -> General Settings -> Miscellaneous -> Date/Time -> NTP.

Check the enable box and enter the IP address of the global NTP server in the *NTP Server:* Click the submit button to apply the configuration. Navigate back to the Time Zone page and select the applicable time zone and click submit to apply the configuration.

Step 3 Verify the NTP configuration on the active and standby central manager by issuing the **show running-config** command via the CLI.

Step 4 Verify the CM clock is in sync with the global NTP server configured on your central manager.

Step 5 The central manager is now synchronized to a global NTP server. The rest of the devices can now be synchronized to central manager.

Navigate in the GUI as follows:

Click Devices -> Device Groups -> All Device Groups -> General Settings -> Miscellaneous -> Date/Time -> NTP.

Check the enable box and enter the IP address of the both the active and standby CM in the *NTP Server:* field and click the submit button to apply the configuration. Navigate back to the *Time Zone* page and select the applicable time zone and click submit to apply the configuration.

Step 6 To push this configuration out to all the device look to the right of the part of the page near the top that reads "Time Zone Settings for Device Group, AllDevicesGroup."

Click the last icon, which should have a popup box that reads *Force Settings on all Devices in Group*. You will see a popup that lists the devices that the configuration will be sent to. Click OK to configure.

Step 7 Verify the configuration on each device by issuing the **show running-config** command via the CLI on each device configured in the *AllDevicesGroup*.

The IP address should be that of the CM (101.1.33.4).

Step 8 Verify that each device is now synchronized to the CM by issuing the **show ntp status** and **show clock** commands.

Each device should have NTP enabled and the CM IP address as the server list. All the times should be in sync.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the central manager to be synchronized with the lab NTP server.
- We expect all WAE devices in the network to accept the NTP configuration by using the GUI.
- We expect the WAE devices in the network to synchronize to the central manager clock.

Results

NTP Configuration and Functionality passed.

Reliability

Reliability of network devices is essential for keeping the network functional. WAEs reliability testing included reloading the devices and verifying the configuration was restored after boot up.

This section contains the following topics:

Central Manager Reload WAE512

This test verified that after a reload, the configuration on a Central Manager WAE was the same as before the reload. The running configuration is verified and then copied to the startup configuration. The device is reloaded when it comes back online it is verified that there are no differences between the pre/post reload configuration, the GUI is again accessible, and all devices show up as online.

Test Procedure

The procedure used to perform the Central Manager Reload WAE512 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the configuration on the DUT with the **show running-config** command.

Step 3 Copy the running configuration to the startup configuration with the **copy running-config startup-config** command.

Step 4 Verify the startup configuration with the **show startup-config** command.

Step 5 Reload the WAE with the **reload** command.

Step 6 After the reload verify that the WAE is configured the same as before the reload with the **show running-config** command.

Step 7 Verify IP connectivity is re-established to both management server and out-of-band networks.

Step 8 Verify that you can open up a browser to the central manager device GUI and that all devices show online as their status.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE device to come online after the reload.
- We expect that the configuration will be the same after the reload.
- We expect that the device GUI will be accessible and that all devices will show up as online.

Results

Central Manager Reload WAE512 passed.

Core Reload WAE7326

This test verified that after a reload, the configuration on a Core WAE was the same as before the reload. The running configuration is verified and then copied to the startup configuration. The device is reloaded when it comes back online it is verified that there are no differences between the pre/post reload configuration. Acceleration via the device is validated after the reload.

Test Procedure

The procedure used to perform the Core Reload WAE7326 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the configuration on the DUT with the **show running-config** command.

Step 3 Copy the running configuration to the startup configuration with the **copy running-config startup-config** command.

Step 4 Verify the startup configuration with the **show startup-config** command.

Step 5 Reload the WAE with the **reload** command.

Step 6 After the reload verify that the WAE is configured the same as before the reload with the **show running-config** command.

Step 7 Verify IP connectivity is re-established to both management server and out-of-band networks.show running-config

Step 8 Verify the WAE is once again acclerating traffic by issuing the **show tfo connection summary** command.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE device to come online after the reload.
- We expect that the configuration will be the same after the reload.
- We expect the WAE to accelerate traffic after the reload.

Results

Core Reload WAE7326 passed.

Edge Reload WAE502

This test verified that after a reload, the configuration on a Edge WAE was the same as before the reload. The running configuration is verified and then copied to the startup configuration. The device is reloaded when it comes back online it is verified that there are no differences between the pre/post reload configuration. Acceleration via the device is validated after the reload.

Test Procedure

The procedure used to perform the Edge Reload WAE502 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the configuration on the DUT with the **show running-config** command.

Step 3 Copy the running configuration to the startup configuration with the **copy running-config startup-config** command.

Step 4 Verify the startup configuration with the **show startup-config** command.

Step 5 Reload the WAE with the **reload** command.

Step 6 After the reload verify that the WAE is configured the same as before the reload with the **show running-config** command.

Step 7 Verify IP connectivity is re-established to both management server and out-of-band networks.

Step 8 From Branch 3 initiate an FTP file transfer from an FTP server in DCB. Verify the file is opimized with the **show tfo connection server-ip 101.1.33.5**, **show statistics tfo**, **show statistics tfo savings**, and **show statistics dre** commands.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE device to come online after the reload.
- We expect that the configuration will be the same after the reload.
- We expect the WAE to accelerate traffic after the reload.

Results

Edge Reload WAE502 failed. The following failures were noted: CSCsi69388 and CSCsi75538.

Edge Reload WAE512

This test verified that after a reload, the configuration on a Edge WAE was the same as before the reload. The running configuration is verified and then copied to the startup configuration. The device is reloaded when it comes back online it is verified that there are no differences between the pre/post reload configuration. Acceleration via the device is validated after the reload.

Test Procedure

The procedure used to perform the Edge Reload WAE512 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the configuration on the DUT with the **show running-config** command.

Step 3 Copy the running configuration to the startup configuration with the **copy running-config startup-config** command.

Step 4 Verify the startup configuration with the **show startup-config** command.

Step 5 While monitoring the console reload the WAE with the **reload** command. Continue monitoring the device until the reload is complete.

Step 6 After the reload, verify that the WAE is configured the same as before the reload with the **show running-config** command.

Step 7 Verify IP connectivity is reestablished to both management server and out-of-band networks.

Step 8 Initiate an FTP connection from a client at the branch and verify that the connection is optimized by issuing the **show tfo connection summary** command.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the WAE device to come online after the reload.
- We expect the configuration to be the same after the reload.
- We expect the WAE to accelerate traffic after the reload.

Results

Edge Reload WAE512 passed.

Upgrade

The compact Flash on the SSL Services Module has two bootable partitions: application partition (AP) and maintenance partition (MP). By default, the application partition boots every time. The application partition contains the binaries necessary to run the SSL image. The maintenance partition is booted if you need to upgrade the application partition.

You can upgrade both the application software and the maintenance software. However, you are not required to upgrade both images at the same time. Refer to the release notes for the SSL Services Module for the latest application partition and maintenance partition software versions.

The entire application and maintenance partitions are stored on the FTP or TFTP server. The images are downloaded and extracted to the application partition or maintenance partition, depending on which image is being upgraded.

To upgrade the application partition, change the boot sequence to boot the module from the maintenance partition. To upgrade the maintenance partition, change the boot sequence to boot the module from the application partition. Set the boot sequence for the module by using the supervisor engine CLI commands. The maintenance partition downloads and installs the application image. The supervisor engine must be executing the run-time image to provide network access to the maintenance partition.

Before starting the upgrade process, you will need to download the application partition image or maintenance partition image to the TFTP server.

A TFTP or FTP server is required to copy the images. The TFTP server should be connected to the switch, and the port connecting to the TFTP server should be included in any VLAN on the switch.

This section contains the following topics:

Core CLI Upgrade WAE612

The WAE devices can be upgraded via the Central Mangager GUI or via the device CLI. This test verified the ability of a WAE-612 running in edge application-accelerator mode to be upgraded to WAAS version under test via the CLI without error. The configuration was logged before and after the upgrade to verify that no discrepancies were seen after the upgrade.

Test Procedure

The procedure used to perform the Core CLI Upgrade WAE612 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On wae-branch1-612-1 issue the **show device-mode current** command to verify the WAE is running in application-accelerator mode.

Step 3 Issue the **show version** command to identify the version of software running on the WAE.

Step 4 Issue the **show running-config** command and log the WAE configuration to a text file.

Step 5 After downloading the new version of software to a FTP server issue the **copy ftp install 10.0.10.10 / WAAS-4.0.11.34-K9.bin** command to download and install the software on the WAE.

Step 6 Issue the **show flash** command to verify the WAE will boot with the new version of code after a reload.

Step 7 While monitoring the console issue the **copy running-config startup-config** and **reload** commands to first save the configuration and then reload the WAE.

Step 8 Once the WAE has rebooted issue the **show version** command to verify is now running the new version of code.

Step 9 Issue the **show running-config** command once again and log it to a text file. Verify that the configuration has not changed since the pre-test log was created.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the edge application-accelerator WAE will upgrade without error.
- We expect that the configuration will be the same after the upgrade.

Results

Core CLI Upgrade WAE612 passed.

Edge GUI Upgrade WAE512

The WAE devices can be upgraded via the Central Mangager GUI or via the device CLI. This test verified the ability of a WAE-512 running in edge application-accelerator mode to be upgraded to WAAS version under test via the CLI without error. The configuration was logged before and after the upgrade to verify that no discrepancies were seen after the upgrade.

Test Procedure

The procedure used to perform the Edge GUI Upgrade WAE512 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On wae-branch1-512-1 issue the **show device-mode current** command to verify the WAE is running in central-manager mode.

Step 3 Issue the **show version** command to identify the version of software running on the WAE.

Step 4 Issue the **show running-config** command and log the WAE configuration to a text file.

Step 5 From wae-server-1 log into the central manager GUI at URL: https://101.1.33.4:8443/. Navigate as follows:

Device -> wae-branch1-512-1 -> Update Software -> Edit Software Files -> Create new software file

Specify the software file URL, username, password, and software version. Check the auto-reload box and then click Submit.

Step 6 From the central manger GUI navigate as follows:

Devices -> wae-branch1-512-1 -> Update Software

Click the submit button to initialize the upgrade and monitor the console as the device reloads and log the output to a text file.

Step 7 Once the WAE has rebooted issue the **show version** command to verify it is now running the new version of code.

Step 8 Issue the **show running-config** command once again and log it to a text file. Verify that the configuration has not changed since before the upgrade.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the edge application-accelerator WAE will upgrade without error.
- We expect that the configuration will be the same after the upgrade.

Results

Edge GUI Upgrade WAE512 passed.

WAFS

Cisco Wide Area File Services (WAFS) software overcomes WAN latency and bandwidth limitations with proprietary Cisco optimization technologies, offering users at branch offices a LAN-like experience when accessing the centralized files over the WAN. This facilitates the consolidation of all branch-office data into central file servers in your data center. WAAS contains a full implementation of the industry-leading WAFS product. The test performed verified the setup of WAFS on the WAAS software platform.

This section contains the following topics:

WAFS Configuration Verification

Cisco WAFS software overcomes WAN latency and bandwidth limitations with proprietary Cisco optimization technologies, offering users at branch offices a LAN-like experience when accessing the centralized files over the WAN. In WAAS 4.0.7 and later CIFS auto-discovery is enabled, which enables WAAS to automatically detect CIFS file servers for optimization using the WAFS Application Optimizer (AO). By automatically discovering file servers for optimization, WAAS administrators are no longer required to manually define each file server for optimization.

This test verified that the WAFS configuration validated for the WAAS network. The configuration of the appropriate system and services was verified. Even though the auto-discovery feature was enabled, a file server was still defined for verification because in some cases this may be preferable. The WAFS policies were verified within the central manager.

Test Procedure

The procedure used to perform the WAFS Configuration Verification test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 From wae-server-1 open an Internet Explorer browser to the central manager.

Step 3 Verify that the WAFS core cluster is created (CM > Devices > Device Groups > Core WAFS Cluster).

Step 4 Verify that the WAFS core WAE is assigned to this cluster (CM > Devices > Device Groups > Core WAFS Cluster > Members, also CM > Devices > Core WAE > File Services > Core Configuration).

Step 5 Verify that the WAFS core service is started on the WAFS core WAE (WAFS core device GUI).

Step 6 Ensure that the file server is configured (CM > Services > File > File Servers > waas-server-1.dcap.com).

Step 7 Verify that the file server is resolvable from the WAFS core cluster (CM > Services > File > File Server > wae-server-1.dcap.com > Assign Core Clusters > Resolve).

Step 8 Verify that the WAFS edge WAE has the edge service enabled and the correct interception and port configuration is applied (CM > Devices > (WAE) > File Services > Edge Configuration).

Step 9 Verify that the connectivity directive is defined (CM > Services > File > Connectivity).

Step 10 Verify that the connectivity directive lists the file server as exported with a box (CM > Services > File > Connectivity > waas-server-1 > File Server Settings).

Step 11 Verify that the connectivity directive lists the appropriate edge devices or groups (CM > Services > File > Connectivity > wae-server-1 > Assign Edge Devices).

Verify that each edge WAE has a green check next to it and that the status is online.

Step 12 Verify that the WAN parameters in the connectivity directive are accurate (CM > Services > File > Connectivity > wae-server-1 > WAN Utilization).

The WAN Defaults are: Maximum allocated bandwidth: 1544 kbps Minimum round-trip delay: 80 ms

Step 13 Verify that the WAFS accept policy is applied against a device group (CM > Devices > Device Group > All Devices > Acceleration > Policies > Definition).

Step 14 Verify that the WAFS transport policy is assigned to the application WAFS, application classifier is set to CIFS, and action is set to full optimization.

Step 15 Verify that the WAFS transport policy is applied against the same device group (CM > Devices > Device Group > All Devices > Acceleration > Policies > Definition).

Step 16 Verify the WAFS map adapter configuration on the WAE devices by issuing the **show running-config** command.

Step 17 Verify that the WAFS edge and WAFS core WAE's have established optimized connections between them using the **show tfo connection summary** commands. Look for connections using server-port 4050.

For more detailed statistics issue, the **show tfo connection server-port 4050** command.

Step 18 Verify in the WAE Device GUI for both the edge and core that they are connected. Visit WAFS Edge > Monitoring and WAFS Core > Monitoring and validate that connection data is being exchanged. A green check mark should appear.

Step 19 Stop background scripts to collect final status of network devices and analyze for error.

Step 20 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect a core cluster to be created.
- We expect a core WAE device to be assigned to the core cluster.
- We expect the core service to be started on the core WAE.
- We expect a file server to be configured.
- We expect the file server name to be resolvable.

- We expect the edge service to be started on the edge WAE.
- We expect the connectivity directive lists the file server as exported.
- We expect the connectivity directive lists the edge devices.
- We expect the WAFS policies to be configured on the core and edge devices.
- We expect the WAFS accept/transport policies to be configured and enabled.

Results

WAFS Configuration Verification passed.

WCCPv2

WCCPv2 is a Cisco-developed content-routing technology that enables you to integrate content engines, such as Wide-Area Application Engines, into your network infrastructure. It is used for transparent interception and redirection of application and file traffic in branch offices and data centers to the local WAE. The tests performed verified the configuration and functionality of WCCPv2 in the WAAS/DCAP topology.

This section contains the following topics:

WCCPv2 Basic Configuration on Edge WAE2821

The Web Cache Communication Protocol (WCCP) is a Cisco-developed content-routing technology that allows integration of WAE devices, among other content engines, into your network infrastructure. The WAE devices are logically deployed using WCCPv2, providing transparent network interception and redirection of packets, high-availability clustering, and load sharing. Once the WAE devices have joined the service group with the router, the router will monitor traffic for flows that should be forwarded to the WAE instead of the original destination. With WCCPv2, up to 32 WAEs can join a service group with up to 32 routers.

This test verified the basic configuration and functionality of WCCPv2 on a Branch ISR. The ingress LAN and WAN port configuration are first verified. The ingress ports from the WAE device(s) configuration(s) are then verified. Finally the Assigned Hash info for service 61 and 62 is verified. In this test the branch device is a 3845 ISR.

Test Procedure

The procedure used to perform the WCCPv2 Basic Configuration on Edge WAE2821 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the WCCP configuration on the core device with the **show running-config** command.

WCCP version 2 is enabled by default. Verify ip wccp 61 and ip wccp 62 are configured globally.

Step 3 On the ingress interface from the WAN verify that **ip wccp 62 redirect in** is configured with the **show running config interface GigabitEthernet0/0.32** command.

Step 4 On the ingress interface from the LAN verify that **ip wccp 61 redirect in** is configured with the **show running config interface GigabitEthernet0/0.30** command.

Step 5 On the interface connecting the WAE device verify that ip wccp redirect exclude in is configured with the show running-config interface integrated-Service-Engine 1/0 command.

Step 6 Verify WCCPv2 is running and that services 61 and 62 are enabled with the **show ip wccp** command.

For services 61 and 62 the *Number of Service Group Clients* is equal to the number of WAE devices connected to the router and the *Number of Service Group Routers* should be 1.

Note: If no Loobpack address is configured on the router then the highest IP address will serve as the Router ID.

Step 7 Verify the interface configuration for the DUT by issuing the **show ip wccp interfaces** command.

The Input services for the ingress WAN and LAN interfaces should be 1 and Exclude In should be FALSE. For the WAE interface(s) all services should be 0 and Exclude In should be TRUE.

Step 8 Verify that the WAE device(s) are correctly seen by issuing **show ip wccp *service*** detail command for service 61 and 62.

The Assigned Has Info depends on the number of WAE devices seen by the router. Verify that if more than one is present that the hash is correct.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect service 61 redirect-in to be configured on the ingress LAN ports.
- We expect service 62 redirect-in to be configured on the ingress WAN ports.
- We expect exclude-in to be configured on the WAE interface.
- We expect that the Assigned Hash Info for services 61 and 62 will be as expected.

Results

WCCPv2 Basic Configuration on Edge WAE2821 passed.

WCCPv2 Configuration and Functionality on Core Sup720

WCCP is a Cisco-developed content-routing technology that allows integration of WAE devices, among other content engines, into your network infrastructure. The WAE devices are logically deployed using WCCPv2, providing transparent network interception and redirection of packets, high-availability clustering, and load sharing. Once the WAE devices have joined the service group with the router, the router will monitor traffic for flows that should be forwarded to the WAE instead of the original destination. With WCCPv2, up to 32 WAEs can join a service group with up to 32 routers.

This test verified the basic configuration and functionality of WCCPv2 on a Supervisor 720. The ingress LAN and WAN port configuration were first verified. The ingress ports from the WAE devices configurations were then verified. Finally the Assigned Hash information for service 61 and 62 was verified.

Test Procedure

The procedure used to perform the WCCPv2 Configuration and Functionality on Core Sup720 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that WCCPv2 services 61 and 62 are enabled globally on the core device with the **show running-config** command.

WCCP version 2 is enabled by default. Verify IP WCCP 61 and IP WCCP 62 are configured globally.

Step 3 On the ingress ports from the WA, verify that **ip wccp 62 redirect in** is configured with the **show-running config interface *interface*** command.

Step 4 On the ingress ports from the aggregation layer of the Data Center LAN, verify that **ip wccp 62 redirect in** is configured with the **show running-config interface *interface*** command.

Step 5 The Cisco Catalyst 6500 can not handle the **ip wccp redirect exclude in** commands. Configuring it will cause packets to be processed in hardware. Verify the interface connecting the WAE to the switch is not configured with this command.

Because inbound redirection is used on the incoming WAN and LAN interfaces, no exclude statement is necessary.

Step 6 Verify the routers loopback address with the **show interfaces loopback 0** command.

This will be the WCCP router ID address.

Step 7 Verify WCCPv2 is running and that services 61 and 62 are enabled with the **show ip wccp 61** and **show ip wccp 62** commands.

In this case the Router Identifier is the routers loopback address and the Protocol Version is 2.0. For services 61 and 62 the Number of Cache Engines is equal to the number of WAE devices connected to the router and the Number of routers should be one.

Step 8 Verify that the WAE devices are correctly seen by issuing the **show ip wccp *service*** detail command for service 61 and 62.

The Assigned Hash Info depends on the number of WAE devices seen by the router. Verify more than one is present, and that the hash is correct.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect service 61 redirect-in to be configured on the ingress LAN ports.
- We expect service 62 redirect-in to be configured on the ingress WAN ports.
- We expect the Assigned Hash Info for services 61 and 62 to be as expected.

Results

WCCPv2 Configuration and Functionality on Core Sup720 passed.

WCCPv2 Configuration and Functionality on Core WAE7326

WCCP is a Cisco-developed content-routing technology that allows integration of WAE devices, among other content engines, into your network infrastructure. The WAE devices are logically deployed using WCCPv2, providing transparent network interception and redirection of packets, high-availability clustering, and load sharing.

This test verified the basic configuration and functionality of WCCPv2 on a core WAE device. The core device was a WAE-7326.

Test Procedure

The procedure used to perform the WCCPv2 Configuration and Functionality on Core WAE7326 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the WCCP configuration on the core devices with the **show running-config** command.

WCCP version 2 should be configured and the IP address of gateway to the WCCP router should be configured as the IP address for router-list 1. TCP promiscuous mode service with L2 redirect should also be turned on with router-list 1 assigned.

Step 3 Verify that WCCP is enabled on the device with the **show wccp status** command.

Step 4 Verify that the WCCP router is seen by issuing the **show wccp routers** command.

The WCCP router has service 61 and 62 enabled. The Router ID is the loopback address of the WCCP router. The Sent to IP address is the IP of the gateway to the WCCP router.

Step 5 Verify the file engine list for services 61 and 62 are seen by the WCCP router by issuing the **show wccp file-engines** command.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect WCCP version 2 to be configured.
- We expect the WCCP router list to be configured with the IP address of the gateway to the WCCP router.
- We expect the core device to see the neighboring WCCP router for services 61 and 62.
- We expect no CPU or memory problems.

Results

WCCPv2 Configuration and Functionality on Core WAE7326 passed.

WCCPv2 Configuration and Functionality on Edge WAE 512

WCCP is a Cisco-developed content-routing technology that allows you to integrate WAE devices into your network infrastructure. The Cisco WAAS Network Module is logically deployed using WCCPv2, providing transparent network interception and redirection of packets, high-availability clustering, and load sharing. Alternatively, PBR can be employed to deploy the Cisco WAAS Network Module in a high-availability configuration with failover support.

This test verified the basic configuration and functionality of WCCPv2 on an edge WAE device. In this case the devices were a WAE-512, WAE-612, and a WAE-502.

Test Procedure

The procedure used to perform the WCCPv2 Configuration and Functionality on Edge WAE 512 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the WCCP configuration on the edge devices with the **show running-config** command.

WCCP version 2 should be configured and the gateway to the WCCP router should configured as the IP address for router-list 1. TCP promiscuous mode service should also be enabled and router-list 1 assigned.

Step 3 Verify that the WCCP router is seen by issuing the **show wccp routers** command.

The WCCP router has service 61 and 62 enabled. The Router ID is either the loopback or the highest IP address on the WCCP router. The Sent to IP address is the IP of the gateway to the WCCP router.

Step 4 Verify the File Engine List for services 61 and 62 are seen by the WCCP router by issuing the **show wccp file-engines** command.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect WCCP version 2 to be configured.
- We expect the WCCP router list to be configured with the IP address of the gateway to the WCCP router.
- We expect the core device to see the neighboring WCCP router for services 61 and 62.

Results

WCCPv2 Configuration and Functionality on Edge WAE 512 passed.

WCCPv2 Configuration and Functionality on Edge WAE3845

WCCP is a Cisco-developed content-routing technology that allows integration of WAE devices, among other content engines, into your network infrastructure. The WAE devices are logically deployed using WCCPv2, providing transparent network interception and redirection of packets, high-availability clustering, and load sharing. Once the WAE devices have joined the service group with the router, the router will monitor traffic for flows that should be forwarded to the WAE instead of the original destination. With WCCPv2, up to 32 WAE's can join a service group with up to 32 routers.

This test verified the basic configuration and functionality of WCCPv2 on a Branch ISR. The ingress LAN and WAN port configuration were first verified. The ingress ports from the WAE devices configuration(s) were then verified. Finally the Assigned Hash information for service 61 and 62 was verified. In this test the branch device was a 3845 ISR.

Test Procedure

The procedure used to perform the WCCPv2 Configuration and Functionality on Edge WAE3845 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the WCCP configuration on the core device with the **show running-config** command.

WCCP version 2 is enabled by default. Verify IP WCCP 61 and IP WCCP 62 are configured globally.

Step 3 On the ingress interface from the WAN verify that **ip wccp 62 redirect in** is configured with the **show running-config interface GigabitEthernet0/0.12** command.

Step 4 On the ingress interface from the LAN verify that **ip wccp 61 redirect in** is configured with the **show running-config interface GigabitEthernet0/0.10** command.

Step 5 On the interface connecting the WAE device verify that **ip wccp redirect exclude in** is configured with the **show running-config interface GigabitEthernet0/0.11** command.

Step 6 Verify WCCPv2 is running and that services 61 and 62 are enabled with the **show ip wccp** command.

For services 61 and 62 the Number of Service Group Clients is equal to the number of WAE devices connected to the router and the Number of Service Group Routers should be one.

Note: If no loopback address is configured on the router then the highest IP address will serve as the router ID.

Step 7 Verify the interface configuration for the DUT by issuing the **show ip wccp interfaces** command.

The input services for the ingress WAN and LAN interfaces should be one and Exclude In should be FALSE. For the WAE interface all services should be zero and Exclude In should be TRUE.

Step 8 Verify that the WAE devices are correctly seen by issuing the **show ip wccp *service*** detail command for service 61 and 62.

The Assigned Has Info depends on the number of WAE devices seen by the router. Verify that one is present, and that the hash is correct.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect service 61 redirect-in to be configured on the ingress LAN ports.
- We expect service 62 redirect-in to be configured on the ingress WAN ports.
- We expect exclude-in to be configured on the WAE interface.
- We expect the Assigned Hash Info for services 61 and 62 to be as expected.

Results

WCCPv2 Configuration and Functionality on Edge WAE3845 passed.

CHAPTER 12

Global Site Selector (GSS)

The Global Site Selector (GSS) leverages the Domain Name System (DNS) to provide clients with reliable and efficient content services. Domain-to-IP address mapping is performed, with consideration to the availability, location and load of content servers. Using the GSS in combination with Cisco's Content Services Switch (CSS), Cisco's Catalyst 6000 Content Switching Module (CSM) and the Application Control Engine (ACE) allows users to create Global Server Load Balancing (GSLB) networks.

The GSS provides configuration and monitoring services through a central configuration manager, the Global Site Selector Manager (GSSM), and through a CLI that is available on each GSS. Configuration for a GSS network is mostly identical on all devices (global config model) and is entered by the user on a single GSS (central configuration model). For standard features the customer may choose to create a network of up to 8 GSSs with global/central configuration. The customer may instead choose to configure and monitor individual devices (local configuration model), in which case the GUI runs independently on each GSS and configuration is not shared.

In the DCAP 4.0 tests, four GSS devices were used in the entire GSS network across both data centers.

Two GSS devices were installed at each data center. A primary GSSM was installed in DCa and a secondary GSSM was installed in DCb.

The GSS receives DNS queries from client DNS proxies (Local D-Proxy servers are installed at each branch location in the DCAP 4.0 test topology which NS Forwards the DNS queries to the GSS devices), and matches these requests with a user-defined set of DNS rules on each GSS. A match on a DNS rule provides the list of 1st, 2nd and 3rd choice sets of answers that should be considered for the request.

Distributed Denial of Service (DDoS) attacks are designed to deny legitimate users access to a specific computer or network resources. These attacks are originated by malicious attackers who send several thousand spoofed DNS requests to a target device. The target then treats these requests as valid and returns the DNS replies to the spoofed recipient (that is, the victim). Since the target is busy replying to the attackers, it drops valid DNS requests from legitimate D-proxies. When the number of requests is in the thousands, the attacker can potentially generate a multi-gigabit flood of DNS replies, thus causing network congestion.

To combat such DDoS problems, the GSS contains a licensed DDoS detection and mitigation module. This DDoS module was added to DCAP 4.0 testing. Security, in regards to DDoS threats, was one of the focal points for added GSLB feature coverage in DCAP 4.0. DCAP 4.0 added feature coverage for 3 major DDoS features of the GSS 2.0, including Anti-Spoofing, Peacetime Learning and Rate-Limiting/DNS Mitigation.

Antispoofing

To overcome spoofed attacks, the GSS uses an anti-spoofing mechanism called Redirect to TCP. This mechanism is used for DNS queries and is also called DNS-proxy. It is based on forcing the client to resend its query using TCP. Once the query arrives in TCP, the GSS uses a challenge/response mechanism to authenticate the source. If the source succeeds with authentication, the GSS sends a TCP reply back. The D-proxy sends a UDP request, while the GSS sends a TC (or truncated) bit. The D-proxy returns on TCP and the GSS then sends the reply on TCP. This feature was tested in DCAP 4.0 by having clients issue recursive DNS requests to one of the four GSS devices for which the source IP address of the UDP datagram was identified on the GSS as a spoofed IP address.

Peacetime Learning

The GSS enforces a limit on the number of DNS packets per second for each individual D-proxy, or an overall global rate limit. It does not enforce a limit for all other traffic. Initially, this limit is the default value. However, the limit can be adjusted during peacetime, or overwritten by configuring either a D-proxy or a group of D-proxies. Once this limit is exceeded, DNS packets are dropped. This feature was tested in DCAP 4.0 by having the GSS learn its threshold during the peacetime learning process. The client sends multiple DNS queries during this peacetime learning state and then the administrator stops the peacetime learning process. Thus, the DNS queries value is recorded in a file. DNS queries are then sent again from the branch clients for which they are denied based on the peacetime learned value.

Rate-Limiting and DNS Mitigation

A reflector attack occurs when the attacker spoofs the IP address of the victim (in this case, the GSS) and sends multiple DNS requests to a DNS server or multiple DNS servers posing as the victim. The amplification effect is based on the fact that small queries can generate larger UDP packets in response and bombard the victim with a high-volume of DNS response traffic. In DCAP 4.0, this feature was tested with both A and NS resource records.

Within a GSS network an answer is a host address which identifies a resource within a network that the GSS can direct a user to respond to a content request. A GSS answers is either a Virtual IP (VIP) address associated with a server load balancer (SLB), a Name Server which can answer queries that the GSS cannot, or a Content Routing Agent (CRA) that use a resolution process called DNS race to send identical and simultaneous responses back to a user's D-proxy. In the DCAP 4.0 configuration, the GSS is authoritative for multiple domain names for which the ACE’s and CSM's modules provide virtualization for services.

The DNS rule also defines the balancing methods that should be applied for choosing from each set of possible answers, and can be combined with advanced features including checking for answers with the closest network proximity to the client's requesting D-proxy, and use of a sticky database.

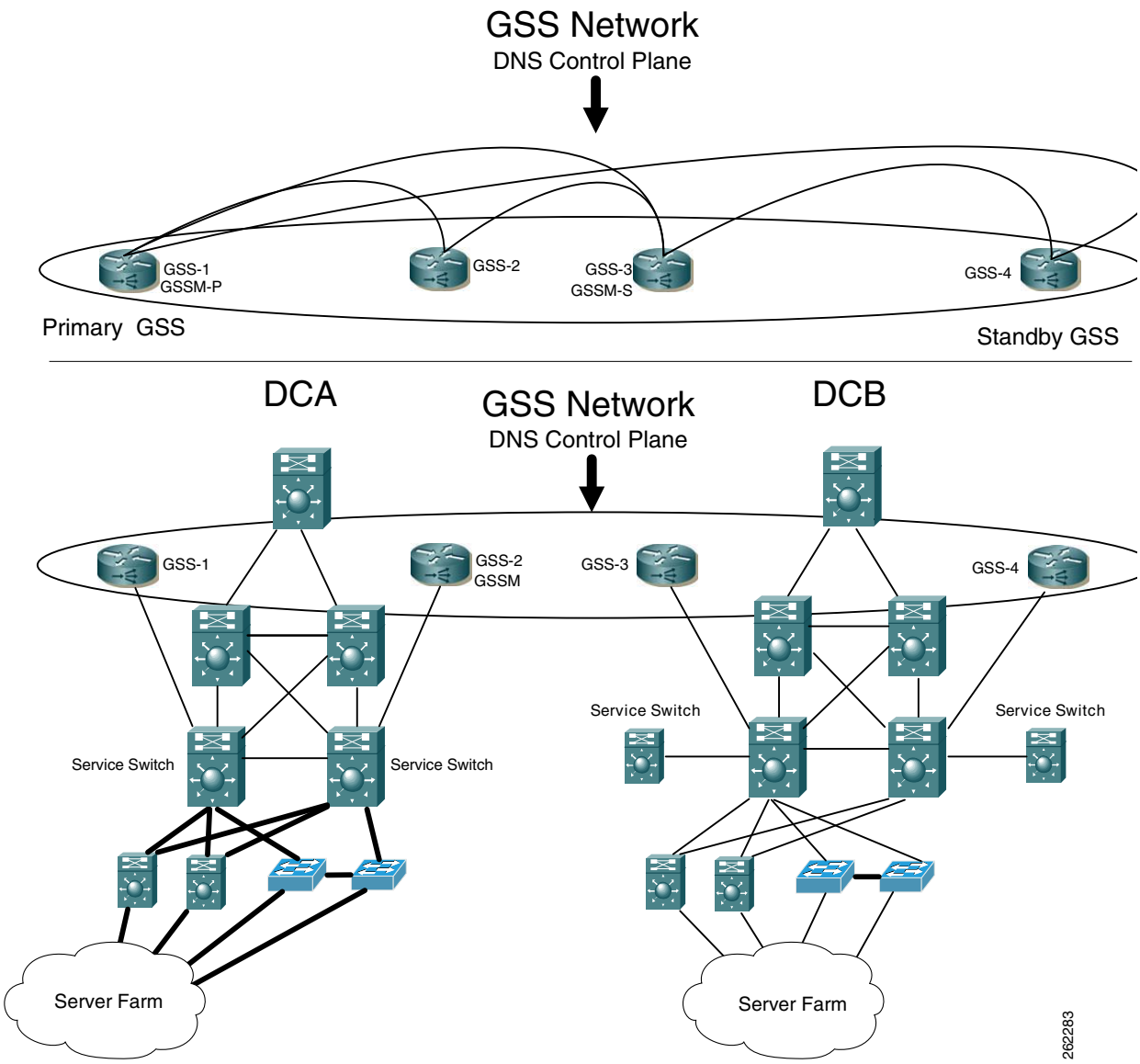
In addition to answering queries directly, the GSS offers the feature of forwarding requests to NS Forwarders, which will return a DNS response packet to the GSS, which in turn returns the exact same response packet to the originally requesting D-proxy. This can be used for any query type on any domain, and is not limited to the record types supported by the GSS. In DCAP 4.0 testing, the NS forwarding feature was used to forward DNS queries for which the GSS was not authoritative for to two top-level DNS servers.

The tests in this chapter focus on the fundamental ability of the GSS working together with existing BIND and Microsoft Name Servers to provide global server load-balancing while providing health monitoring for Oracle applications at each data center through the use of ACE's modules installed in Data Center A and CSMs installed in DCb.

GSS Topology

The GSS devices are integrated into the existing DCAP 4.0 topology (Figure 12-1) along with BIND Name Servers and tested using various DNS rules configured on the GSS. Throughout the testing, the GSS receives DNS queries sourced from client machines as well as via DNS proxies (D-Proxies). The Name Server zone files on the D-Proxies are configured to nsfoward DNS queries to the GSS to obtain authoritative responses. Time-to-Live (TTL) values associated with the various DNS resource records are observed and taken into consideration throughout the testing.

Figure 12-1 DCAP GSS Test Topology



Test Results Summary

Table 12-1 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 12-1 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTS's.

Table 12-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
DDoS, page 12-6	n/a	1. Antispoofing 2. Peacetime Learning 3. Rate-Limiting and DNS Mitigation	
DNS Processing, page 12-10	n/a	1. GSS DNS Reqeust Processing	CSCsk51868
DNS Proximity, page 12-13	n/a	1. Dynamic Proximity (no RESET) Wait Disabled 2. Dynamic Proximity (no RESET) Wait Enabled 3. Dynamic Proximity (with RESET) Wait Disabled (Complete) 4. Dynamic Proximity (with RESET) Wait Disabled 5. Static Proximity Branch 1 and Branch 3 (Complete)	
DNS Sticky, page 12-21	n/a	1. Global Sticky Branch 1 and Branch 3 (Complete)	
Keepalives, page 12-22	n/a	1. GSS Kalap to CSM using VIP (Complete) 2. KAL-AP by TAG—Complete	
LB Methods, page 12-25	n/a	1. LB Methods—Complete	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

DDoS

This section contains the following topics:

Anti Spoofing

The Anti spoofing feature on the GSS is a feature that is configured under the ddos module on the GSS. Anti spoofing will force a client dns request to use a transport of TCP rather than UDP. Provided the GSS is configured for the specific spoofed IP Address, the GSS will receive the UDP based DNS query, then the GSS will ask the client to re-send the dns query using TCP as a transport. The GSS uses a challenge/response mechanism to authenticate the source of the dns query.

Test Procedure

The procedure used to perform the Anti Spoofing test follows:

Step 1 On both client machines, branch1-client-1 and branch3-client-1, ensure the primary and secondary name servers are 10.0.30.2 and 10.0.20.2, respectively.

Step 2 On both client machines, branch1-client-1 and branch3-client-1, flush the DNS resolver cache

Step 3 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 4 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com enable DNS console logging by issuing the command

```
logging disk enable logging disk subsystem dns priority debugging
logging disk subsystem ddos priority debuggingOn all 4 GSS's, dca-gss-1.gslb.dcap.com,
dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, clear the
ddos statistics by issuing the command clear statistics ddos all.
```

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, enable real-time logging by issuing the command **show log follow**.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, restore the factory defaults for ddos by issuing the command **ddos restore-defaults**.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, configure ddos feature for dproxy spoofing by issuing the command **ddos dproxy spoofed 10.0.10.2**,**ddos dproxy spoofed 10.0.20.2**,**ddos dproxy spoofed 10.0.30.2**.

Step 8 On all three branch client machines; branch1-client-1, branch2-client-1, and branch3-client-1 issue dns requests for the domain spoof.gslb.dcap.com

Step 9 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, configure ddos feature to delete the dproxy spoofing by issuing the command **no ddos dproxy spoofed 10.0.10.2**,**no ddos dproxy spoofed 10.0.20.2**,**no ddos dproxy spoofed 10.0.30.2**.

Step 10 On both branch client machines; branch1-client-1 and branch3-client-1 issue dns requests for the domain spoof.gslb.dcap.com

Step 11 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, view the ddos peacetime learning stats by issuing the command **show statistics ddos global**.

Expected Results

The following test results are anticipated:

- We expect that the GSS respond appropriately to dns queries that are being sent via a client from multiple dproxy's (sourece IP address of the dns query) when ddos spoofing is configured on the GSS.
- We expect no CPU or memory problems.

Results

Anti Spoofing passed.

Peace Time Learning

This test verified that the GSS learned the expected source IP address of the dns queries (d-proxy IP Address) when the peacetime learning was enabled. A file created though the peacetime learning process was loaded into the GSS in order to verify proper behavior for new dns queries against the file which was loaded onto the GSS.

Test Procedure

The procedure used to perform the Peace Time Learning test follows:

Step 1 On both client machines, branch1-client-1 and branch3-client-1, ensure the primary and secondary name servers are 10.0.30.2 and 10.0.20.2, respectively.

Step 2 On both client machines, branch1-client-1 and branch3-client-1, flush the DNS resolver cache

Step 3 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 4 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com enable DNS console logging by issuing the command

```
logging disk enable logging disk subsystem dns priority debugging
logging disk subsystem ddos priority debuggingOn all 4 GSS's, dca-gss-1.gslb.dcap.com,
dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, clear the
ddos statistics by issuing the command clear statistics ddos all.
```

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, enable real-time logging by issuing the command **show log follow**.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, restore the factory defaults for ddos by issuing the command **ddos restore-defaults**.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, start ddos peacetime learning by issuing the command **ddos peacetime start**.

Step 8 On the client machine, branch1-client-1, issue 20 dns request for the domain wwwin-oefin.gslb.dcap.com

Step 9 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, stop ddos peacetime learning by issuing the command **ddos peacetime stop**.

Step 10 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, save the ddos database file by issuing the command **ddos peacetime save peace_1**.

Step 11 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, view the ddos peacetime learning stats by issuing the command **show statistics ddos global**.

Step 12 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, view the ddos peacetime learning file (which shows the dproxy IP address's and Peacetime Peak DNS requests per minute) by issuing the command **ddos peacetime show peace_1**.

Step 13 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, apply ddos peacetime learning by issuing the command **ddos peacetime apply overwrite**.

Step 14 On the client machine, branch1-client-1 issue more than 20 dns request for the domain wwwin-oefin.gslb.dcap.com in the same amount of time as was sent in previously.

Step 15 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, view the ddos peacetime learning stats by issuing the command **show statistics ddos global**.

Expected Results

The following test results are anticipated:

- We expect the GSS to learn the expected dns queries via the source d-proxy when peacetime learning is enabled.

Results

Peace Time Learning passed.

Rate Limiting DNS Mitigation

This test will show that the GSS responds appropriately to multiple dns queries sourced from a d-proxy based on the rate at which the queries are sent.

Test Procedure

The procedure used to perform the Rate Limiting DNS Mitigation test follows:

Step 1 On the client machine, branch2-client-1, ensure the primary name server is 10.0.20.2

Step 2 On the client machine, branch2-client-1, flush the DNS resolver cache

Step 3 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 4 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com enable DNS console logging by issuing the command

```
logging disk enable logging disk subsystem dns priority debugging
logging disk subsystem ddos priority debuggingOn all 4 GSS's, dca-gss-1.gslb.dcap.com,
dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, clear the
ddos statistics by issuing the command clear statistics ddos all.
```

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify that the proper logging levels are applied by issuing the command **show logging**.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, enable real-time logging by issuing the command **show log follow**.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, configure the ddos rate-liming values by issuing the commands: **ddos restore-defaults rate-limit global 5 rate-limit 10.0.20.2 5 rate-limit unknown 5 scaling-factor d-proxy 2**

Step 8 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, view the dns ddos rate limiting statitics by issuing the command **show ddos rate-limit 10.0.20.2**.

Step 9 On the client machine, branch1-client-1, issue 10 dns request for the domain wwwin-oefin.gslb.dcap.com

Step 10 On the GSS handled the domain requests, view the ddos stats by issuing the command **show ddos rate-limit 10.0.20.2**.

Step 11 On the client machine, branch2-client-1, issue 10 dns request for the domain wwwin-oefin.gslb.dcap.com directly to the GSS in DCA. In the previous steps, DNS requests were issued to the GSS via the local name server.

Step 12 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, view the dns ddos rate limiting statitics by issuing the command **show ddos rate-limit 10.0.20.3**.

Step 13 On the client machine, branch2-client-1, issue 10 dns request for the domain rate-limit.gslb.dcap.com to the local name server but this time, send a query for an NS record by issuing the command

set type=ns

Step 14 On the GSS that handled the NS record dns query, view the dns ddos rate limiting statitics by issuing the command **show ddos rate-limit 10.0.20.2**.

Step 15 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, disable ddos by issuing the command **no ddos enable**.

Step 16 On the client machine, branch2-client-1, issue 10 dns requests for the domain wwwin-oefin.gslb.dcap.com

Expected Results

The following test results are anticipated:

- We expect that the GSS will enforce the properly configured d-proxy limit on the number of DNS packets per second for each individual D-proxy.
- We expect no CPU or memory problems.

Results

Rate Limiting DNS Mitigation passed.

DNS Processing

This section contains the following topics:

GSS DNS Reqeust Processing

This test verified that the GSS responded property when sent different DNS resource record types. DNS queries were sent from client machines directly to the GSS, and from client machines to NS forwarding name servers (D-proxies) to the GSS.

Test Procedure

The procedure used to perform the GSS DNS Reqeust Processing test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On both client machines, gss-winxp-1 and gss-linux-1, ensure the primary and secondary name servers are 10.0.5.111 and 10.0.5.102, respectively.

Step 3 On both client machines, gss-winxp-1 and gss-linux-1, flush the DNS resolver cache.

Step 4 On both GSS's, dcap-gss-1.gslb.com and dcap-gss-2.gslb.com, force a rotation and deletion of logs by issuing the**rotate-logs** and **rotate-logs delete-rotated-logs** commands.

Step 5 On both GSS's, dcap-gss-1.gslb.com and dcap-gss-2.gslb.com, enable DNS console logging by issuing the **logging disk enable** and **logging disk subsystem dnsserver priority debugging** commands. Ensure DNS logging is enabled by issuing the **show logging** command.

Step 6 On both GSS's, dcap-gss-1.gslb.com and dcap-gss-2.gslb.com, clear the DNS statistics by issuing the **clear statistics dns** command. Ensure the DNS statistics have been cleared by issuing the **show statistics dns global** command.

Step 7 On both GSS's, dcap-gss-1.gslb.com and dcap-gss-2.gslb.com, enable real time logging by issuing the **show log follow** command.

Step 8 Verify both GSS's respond with the expected result. Send a DNS A record query from both clients, gss-winxp-1 and gss-linux-1, to both of the name servers. From gss-linux-1 issue the following commands:

- dig @10.0.5.111 eng.gslb.com. a +qr
- dig @10.0.5.102 eng.gslb.com. a +qr

```
From gss-winxp-1 issue the following commands:
```

nslookup set d2 server 10.0.5.111 eng.gslb.com. server 10.0.5.102 eng.gslb.com

Step 9 On both GSS's, dcap-gss-1.gslb.com and dcap-gss-2.gslb.com, verify the DNS global statistics by issuing the **show statistics dns global** command.

Step 10 Verify both GSS's respond with the expected result for hostnames for which the GSS's are authoritative when responding to a DNS query type of AAAA. Send a DNS AAAA record query from both clients gss-winxp-1 and gss-linux-1, to both of the name servers. From gss-linux-1 issue the following commands:

```
dig @10.0.5.111 eng.gslb.com. aaaa +qr
dig @10.0.5.102 eng.gslb.com. aaaa +qrFrom gss-winxp-1
```

nslookup set d2 set type=aaaa server 10.0.5.111 eng.gslb.com. server 10.0.5.102 eng.gslb.com

Step 11 Verify both GSS's respond with the expected result for host names that the GSS's are not authoritative for when responding to a DNS query type of A. From gss-linux-1 issue the following commands:

```
dig @10.0.5.111 not-here.gslb.com. a +qr
dig @10.0.5.102 not-here.gslb.com. a +qrFrom gss-winxp-1 issue the following commands:
```

nslookup set d2 set type=a server 10.0.5.111 not-here.gslb.com. server 10.0.5.102 not-here.gslb.com.

Step 12 Using the following commands, verify the GSS responds with the correct/valid response when sending DNS A queries for which the GSS is not authoritative for by asking the GSS directly:

dig @101.1.32.11 not-here.gslb.com. a +qr dig @101.1.32.12 not-here.gslb.com. a +qr

Step 13 Verify both GSS's respond with the expected result for host names, that the GSS's are authoritative for the wildcard domain of .*\.gslb\.com. Ensure all other DNS rules on the GSS are suspended. Ask the GSS directly. From gss-linux-1 issue the following commands:

```
dig @101.1.32.11 wildcard.gslb.com. a +qr
dig @101.1.32.12 wildcard.gslb.com. a +qrFrom gss-winxp-1 issue the following commands:
```

nslookup set d2 set type=a server 101.1.32.11 wildcard.gslb.com. server 101.1.32.12 wildcard.gslb.com.

Step 14 Verify the GSS responds with the correct/valid response when sending valid DNS A queries to the GSS for which the GSS is authoritative for the wild card domain of .*\.gslb\.com. Issue the following commands:

dig @10.0.5.111 eng.gslb.com. a dig @10.0.5.102 eng.gslb.com. a

```
Ensure all other DNS rules on the GSS are suspended.
```

Step 15 Verify both GSS's respond with the expected result for hostnames for which the GSS's are authoritative the wild card domain of .*\.gslb\.com, but does not support the resource record type. Ensure all other DNS rules on the GSS are suspended. Send the DNS queries directly to the GSS. From gss-linux-1 issue the following commands:

```
dig @101.1.32.11 wildcard.gslb.com. MX +qr
```

```
dig @101.1.32.12 wildcard.gslb.com. MX +qrFrom gss-winxp-1 issue the following commands:
```
nslookup set d2 set type=mx server 101.1.32.11 wildcard.gslb.com. server 101.1.32.12 wildcard.gslb.com.

Step 16 Verify both GSS's respond with the expected result for hostnames for which the GSS's are authoritative when responding to a DNS query type of A using TCP as a transport rather than UDP. Send the DNS queries directly to the GSS. From gss-linux-1 issue the following commands:

dig @101.1.32.11 eng.gslb.com. a +tcp +qr dig @101.1.32.12 eng.gslb.com. a +tcp +qr

Step 17 Verify both GSS's respond with the expected result for hostnames for which the GSS's are authoritative when responding to a DNS query type A queries and setting the UDP message buffer size (EDNS0 bytes). Send the DNS queries directly to the GSS. From gss-linux-1 issue the following commands:

dig @101.1.32.11 eng.gslb.com a +bufsiz=1024 +qr dig @101.1.32.12 eng.gslb.com a +bufsiz=1024 +qr

Step 18 Verify both GSS's respond with the expected result when NS forwarding DNS type A queries. Send the DNS queries directly to the GSS. From gss-linux-1 issue the following commands:

```
dig @101.1.32.11 send-me-away.gslb.com +qr
dig @101.1.32.12 send-me-away.gslb.com +qrFrom gss-winxp-1 issue the following commands:
```
nslookup set d2 set type=a server 101.1.32.11 send-me-away.gslb.com. server 101.1.32.12 send-me-away.gslb.com.

Step 19 Verify both GSS's respond with the expected result when NS forwarding DNS type a queries using TCP rather than UDP. Send the dns queries directly to the GSS. From gss-linux-1 issue the following commands:

dig @101.1.32.11 send-me-away.gslb.com +tcp +qr dig @101.1.32.12 send-me-away.gslb.com +tcp +qr

Step 20 Verify both GSS's respond with the expected result when NS forwarding DNS type MX queries. Send the DNS queries directly to the GSS. From gss-linux-1 issue the following commands:

```
dig @101.1.32.11 mail.gslb.com mx +qr
dig @101.1.32.12 mail.gslb.com mx +qrFrom gss-winxp-1 issue the following commands:
```
nslookup set d2 set type=mx server 101.1.32.11 mail.gslb.com. server 101.1.32.12 mail.gslb.com.

Step 21 Verify both GSS's respond with the expected result for hostnames for which the GSS's are authoritative when responding to a DNS query type of MX. Send the DNS queries directly to the GSS. From gss-linux-1 issue the following commands:

```
dig @101.1.32.11 eng.gslb.com. mx +qr
dig @101.1.32.12 eng.gslb.com. mx +qrFrom gss-winxp-1
```
nslookup set d2 set type=mx server 101.1.32.11 eng.gslb.com. server 101.1.32.12 eng.gslb.com

Step 22 Verify both GSS's respond with the expected result for hostnames for which the GSS's are authoritative when responding to a DNS query type of any. Send the DNS queries directly to the GSS. From gss-linux-1 issue the following commands:

```
dig @101.1.32.11 eng.gslb.com. any +qr
dig @101.1.32.11 eng.gslb.com. any +qrFrom gss-winxp-1 issue the following commands:
```
nslookup set d2 set type=any server 101.1.32.11 eng.gslb.com. server 101.1.32.11 eng.gslb.com.

Step 23 Stop background scripts to collect final status of network devices and analyze for error.

Step 24 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the GSS to respond to various well formed, RFC-based DNS queries in the proper manner.

Results

GSS DNS Reqeust Processing failed. The following failures were noted: CSCsk51868.

DNS Proximity

This section contains the following topics:

Dynamic Proximity (no RESET) Wait Disabled

This test verified that the GSS responded with the correct answers based on the DRP agent probes and the fact that the the D-proxy does not issue a TCP RST in response to the DRP probe SYN/ACK.

Test Procedure

The procedure used to perform the Dynamic Proximity (no RESET) Wait Disabled test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On both client machines, branch1-client-1 and branch3-client-1, ensure the primary and secondary name servers are 10.0.30.2 and 10.0.20.2, respectively

Step 3 On both client machines, branch1-client-1 and branch3-client-1, flush the DNS resolver cache

Step 4 Ensure on the GSS rule "wwwin-oefin" that proximity is enabled and that WAIT is set to "disabled".

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com enable DNS console logging by issuing the command **logging disk enable** and the command **logging disk subsystem proximity priority debugging** Ensure DNS logging is enabled by issuing the command **show logging**.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com issue the command **show log follow**. Verify you are seeing the correct log output. End the real-time logging by issuing the command **CTR-C**.

Step 8 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com clear the proximity database by issuing the command **proximity database delete all**

Step 9 RDP into the branch name server for branch 1 and branch 3. Open Wireshark or Ethereal on the name server. Start capturing traffic on the DC network interface in order to view the probes that are sourced from the DRP router at each data center to each data center's nameserver.

Step 10 From a GSS's in DCA, test the proximity probing by by issuing the command **proximity probe 10.0.10.2 zone all**, and the command **proximity probe 10.0.30.2 zone all**.

Step 11 Open Wireshark or Ethereal on the name server for which you captured the trace of the DRP probes. Verify the you see SYN/ACK's sent from the DRP router and that you do not see any RST's sent back from the name server.

Step 12 From the same GSS in DCA, verify that a proximity entry has not been added to the GSS database by issuing the command **proximity database dump DUMP_PD format xml**

Step 13 From a GSS in DCB, test the proximity probing by by issuing the command **proximity probe 10.0.10.2 zone all**, and the command **proximity probe 10.0.30.2 zone all**.

Step 14 From the same GSS in DCB, verify that a proximity entry has not been added to the GSS database by issuing the command **proximity database dump DUMP_PD format xml**

Step 15 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com clear the proximity statitics by issuing the command **clear statistics proximity**

Step 16 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify all CSM VIP's for clause #2 in GSS rule wwwin-oefin are online by issuing the command **show statistics keepalive tcp list**.

Step 17 On both client machines, branch1-client-1 and branch3-client-1, ensure the correct dns behavior and the correct resource record is returned to the client by issuing nslookup from both clients.

Step 18 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com view the realtime logging by issuing the command **show log follow**

Step 19 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com verify the proximity lookup statitics by issuing the command **show statistics proximity lookup**

Step 20 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com issue the command **show log | grep PRX** and **show log | grep Measurement**. in order to verify you are seeing the correct log output.

Step 21 Stop background scripts to collect final status of network devices and analyze for error.

Step 22 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the GSS will properly service dns requests for a dns rule that is enabled for dynamic proximity while the name server being probed does not issue a TCP RST. The next clause in the dns rule should be matched.

Results

Dynamic Proximity (no RESET) Wait Disabled passed.

Dynamic Proximity (no RESET) Wait Enabled

This test verified that the GSS responded with the correct answers based on the DRP agent probes and the fact that the the D-proxy does not issue a TCP RST in response to the DRP probe SYN/ACK.

Test Procedure

The procedure used to perform the Dynamic Proximity (no RESET) Wait Enabled test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On both client machines, branch1-client-1 and branch3-client-1, ensure the primary and secondary name servers are 10.0.30.2 and 10.0.20.2, respectively

Step 3 On both client machines, branch1-client-1 and branch3-client-1, flush the DNS resolver cache

Step 4 Ensure on the GSS rule "wwwin-oefin" that proximity is enabled and that that WAIT is set to "enabled".

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com enable DNS console logging by issuing the command **logging disk enable** and the command **logging disk subsystem proximity priority debugging** Ensure DNS logging is enabled by issuing the command **show logging**.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com issue the command **show log follow**. Verify you are seeing the correct log output. End the real-time logging by issuing the command **CTR-C**.

Step 8 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com clear the proximity database by issuing the command **proximity database delete all**

Step 9 RDP into the branch name server for branch 1 and branch 3. Open Wireshark or Ethereal on the name server. Start capturing traffic on the DC network interface in order to view the probes that are sourced from the DRP router at each data center to each data center's nameserver.

Step 10 From a GSS's in DCA, test the proximity probing by by issuing the command **proximity probe 10.0.10.2 zone all**, and the command **proximity probe 10.0.30.2 zone all**.

Step 11 Open Wireshark or Ethereal on the name server for which you captured the trace of the DRP probes. Verify the you see SYN/ACK's sent from the DRP router and that you do not see any RST's sent back from the name server.

Step 12 From the same GSS in DCA, verify that a proximity entry has not been added to the GSS database by issuing the command **proximity database dump DUMP_PD1 format xml**

Step 13 From a GSS in DCB, test the proximity probing by by issuing the command **proximity probe 10.0.10.2 zone all**, and the command **proximity probe 10.0.30.2 zone all**.

Step 14 From the same GSS in DCB, verify that a proximity entry has not been added to the GSS database by issuing the command **proximity database dump DUMP_PD2 format xml**

Step 15 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com clear the proximity statitics by issuing the command **clear statistics proximity**

Step 16 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify all CSM VIP's for clause #2 in GSS rule wwwin-oefin are online by issuing the command **show statistics keepalive tcp list**.

Step 17 On both client machines, branch1-client-1 and branch3-client-1, ensure the correct dns behavior and the correct resource record is returned to the client by issuing nslookup from both clients.

Step 18 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com view the realtime logging by issuing the command **show log follow**

Step 19 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com verify the proximity lookup statitics by issuing the command **show statistics proximity lookup**

Step 20 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com issue the command **show log | grep PRX** and **show log | grep Measurement**. in order to verify you are seeing the correct log output.

Step 21 Stop background scripts to collect final status of network devices and analyze for error.

Step 22 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the GSS will properly service dns requests for a dns rule that is enabled for dynamic proximity while the name server being probed does not issue a TCP RST. The next clause in the dns rule should be matched.

Results

Dynamic Proximity (no RESET) Wait Enabled passed.

Dynamic Proximity (with RESET) Wait Disabled (Complete)

This test verified that the GSS responded with the correct answers based on the DRP agent probes and the fact that the the D-proxy does issue a TCP RST in response to the DRP probe SYN/ACK.

Test Procedure

The procedure used to perform the Dynamic Proximity (with RESET) Wait Disabled (Complete) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On both client machines, branch1-client-1 and branch3-client-1, ensure the primary and secondary name servers are 10.0.30.2 and 10.0.20.2, respectively

Step 3 On both client machines, branch1-client-1 and branch3-client-1, flush the DNS resolver cache

Step 4 Ensure on the GSS rule "wwwin-oefin" that proximity is enabled and that that WAIT is set to "enabled".

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com enable DNS console logging by issuing the command **logging disk enable** and the command **logging disk subsystem proximity priority debugging** Ensure DNS logging is enabled by issuing the command **show logging**.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com issue the command **show log follow**. Verify you are seeing the correct log output. End the real-time logging by issuing the command **CTR-C**.

Step 8 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com clear the proximity database by issuing the command **proximity database delete all**

Step 9 SSH into "wan-emulator-2". Start capturing traffic on the DC network interface in order to view the probes that are sourced from the DRP router at each data center.

Step 10 From a GSS's in DCA, test the proximity probing by by issuing the command **proximity probe 10.0.20.4 zone all**.

Step 11 Viewing TCPDUMP on the "wan-emulator-2" host, verify that you see SYN/ACK's sent from the DRP router and that the "wan-emulator-2" host is responding to the SYN/ACK with a RESET.

Step 12 From the same GSS in DCA, verify that a proximity entry has not been added to the GSS database by issuing the command **proximity database dump DUMP_PD format xml**

Step 13 From a GSS in DCB, test the proximity probing by by issuing the command **proximity probe 10.0.20.4 zone all**.

Step 14 From the same GSS in DCB, verify that a proximity entry has not been added to the GSS database by issuing the command **proximity database dump DUMP_PD format xml**

Step 15 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com clear the proximity statitics by issuing the command **clear statistics proximity**

Step 16 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify all CSM VIP's for clause #2 in GSS rule wwwin-oefin are online by issuing the command **show statistics keepalive tcp list**.

Step 17 On both client machines, wan-emulator-2 and branch3-client-1, ensure the correct dns behavior and the correct resource record is returned to the client by issuing nslookup from branch3-client-1 and by using dig from "wan-emulator-2" for the domain wwwin-oefin.gslb.dcap.

Step 18 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com view the realtime logging by issuing the command **show log follow**

Step 19 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com verify the proximity lookup statitics by issuing the command **show statistics proximity lookup**

Step 20 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com issue the command **show log | grep PRX** and **show log | grep Measurement**. in order to verify you are seeing the correct log output.

Step 21 Stop background scripts to collect final status of network devices and analyze for error.

Step 22 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect no CPU or memory problems.

Results

Dynamic Proximity (with RESET) Wait Disabled (Complete) passed.

Dynamic Proximity (with RESET) Wait Disabled

This test verified that the GSS responded with the correct answers based on the DRP agent probes and the fact that the the D-proxy does issue a TCP RST in response to the DRP probe SYN/ACK.

Test Procedure

The procedure used to perform the Dynamic Proximity (with RESET) Wait Disabled test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On both client machines, branch1-client-1 and branch3-client-1, ensure the primary and secondary name servers are 10.0.30.2 and 10.0.20.2, respectively

Step 3 On both client machines, branch1-client-1 and branch3-client-1, flush the DNS resolver cache

Step 4 Ensure on the GSS rule "wwwin-oefin" that proximity is enabled and that that WAIT is set to "disabled".

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com enable DNS console logging by issuing the command **logging disk enable** and the command **logging disk subsystem proximity priority debugging** Ensure DNS logging is enabled by issuing the command **show logging**.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com issue the command **show log follow**. Verify you are seeing the correct log output. End the real-time logging by issuing the command **CTR-C**.

Step 8 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com clear the proximity database by issuing the command **proximity database delete all**

Step 9 SSH into "wan-emulator-2". Start capturing traffic on the DC network interface in order to view the probes that are sourced from the DRP router at each data center.

Step 10 From a GSS's in DCA, test the proximity probing by by issuing the command proximity probe 10.0.20.4 zone all.

Step 11 Viewing TCPDUMP on the "wan-emulator-2" host, verify that you see SYN/ACK's sent from the DRP router and that the "wan-emulator-2" host is responding to the SYN/ACK with a RESET.

Step 12 From the same GSS in DCA, verify that a proximity entry has not been added to the GSS database by issuing the command **proximity database dump DUMP_PD format xml**

Step 13 From a GSS in DCB, test the proximity probing by by issuing the command proximity probe 10.0.20.4 zone all.

Step 14 From the same GSS in DCB, verify that a proximity entry has not been added to the GSS database by issuing the command **proximity database dump DUMP_PD format xml**

Step 15 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com clear the proximity statitics by issuing the command **clear statistics proximity**

Step 16 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify all CSM VIP's for clause #2 in GSS rule wwwin-oefin are online by issuing the command **show statistics keepalive tcp list**.

Step 17 On both client machines, wan-emulator-2 and branch3-client-1, ensure the correct dns behavior and the correct resource record is returned to the client by issuing nslookup from branch3-client-1 and by using dig from "wan-emulator-2" for the domain wwwin-oefin.gslb.dcap.

Step 18 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com view the realtime logging by issuing the command **show log follow**

Step 19 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com verify the proximity lookup statitics by issuing the command **show statistics proximity lookup**

Step 20 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com issue the command **show log | grep PRX** and **show log | grep Measurement**. in order to verify you are seeing the correct log output.

Step 21 Stop background scripts to collect final status of network devices and analyze for error.

Step 22 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect no CPU or memory problems.

Results

Dynamic Proximity (with RESET) Wait Disabled passed.

Static Proximity Branch 1 and Branch 3 (Complete)

This test verified that the GSS responded with the correct answers based on the source address of the D-proxy.

Test Procedure

The procedure used to perform the Static Proximity Branch 1 and Branch 3 (Complete) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the client's branch1-client-1.cisco.com and branch3-client-1.cisco.com, ensure the primary and secondary name server is 10.0.10.2 and 10.0.30.2, respectively.

Step 3 Flush the DNS resolver cache on the client's branch1-client-1.cisco.com and branch3-client-1.cisco.com.

Step 4 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com , force a rotation and deletion of logs by issuing the **rotate-logs** and **rotate-logs delete-rotated-logs** commands.

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com , enable DNS console logging by issuing the **logging disk enable** and **logging disk subsystem dnsserver priority debugging** commands. Ensure DNS logging is enabled by ensuring the **show logging** command.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com , clear the DNS statistics by issuing the **clear statistics dns** command. Ensure the DNS statistics have been cleared by issuing the **show statistics dns global** command.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, enable real time logging by issuing the **show log follow** command.

Step 8 On the GSS, configure the DNS rule "wwwin-oefin" with the source address list of "branch-1-src", and verify that all 4 GSS's respond with the expected result when using both D-proxy name servers. Issue the following commands: nslookup

Step 9 On the GSS, configure the DNS rule "wwwin-oefin" with the source address list of "branch-3-src", and verify both GSS's respond with the expected result when using both D-proxy name servers. Issue the following commands: nslookup

Step 10 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify the DNS global statitics by issuing the **show statistics dns global** command, and verify the source address lists statistics by issuing the **show statistics dns source-address** command.

Step 11 On the GSS, configure the DNS rule "wwwin-oefin" with the source address list of "branch-2-src", and verify all GSS's respond with the expected result when using both D-proxy name servers. Issue the following commands: dig @10.0.5.111 eng.gslb.com. a +qr dig @10.0.5.102 eng.gslb.com. a +qr

Step 12 On the GSS, ensure the DNS rule "wwwin-oefin" with the source address list of "branch-1-src", and verify both GSS's respond with the expected result when using both D-proxy name servers. Point each of the two branch client's (branch1-client-1.cisco.com and branch3-client-1.cisco.com) to one of the 4 GSS's direclty.

Step 13 View the "dns source address" statitics on each GSS the client's asked and verify you are seeing the expected behavior.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the GSS to respond with the correct answers based on the source address of the D-proxy.

Results

Static Proximity Branch 1 and Branch 3 (Complete) passed.

DNS Sticky

This section contains the following topics:

Global Sticky Branch 1 and Branch 3 (Complete)

This test verified that the GSS properly replicated DNS responses to its peer GSS while maintaining affinity based on the source of the D-proxy. VIP's were taken offline in order to ensure the proper answer was provided by the GSS and replicated to its peers.

Test Procedure

The procedure used to perform the Global Sticky Branch 1 and Branch 3 (Complete) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the client's branch1-client-1.cisco.com and branch3-client-1.cisco.com, ensure the primary and secondary name server is 10.0.10.2 and 10.0.30.2, respectively.

Step 3 Flush the DNS resolver cache on client's branch1-client-1.cisco.com and branch3-client-1.cisco.com.

Step 4 On the GSS, enable DNS console logging by issuing the **logging disk enable** and **logging disk subsystem sticky priority debugging** commands. Ensure DNS logging is enabled by ensuring the **show logging** command.

Step 5 On the GSS, force a rotation and deletion of logs by issuing the **rotate-logs** and **rotate-logs delete-rotated-logs** commands.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, clear the DNS statistics by issuing the **clear statistics sticky** command. Ensure the sticky statistics have been cleared by issuing the **show statistics sticky global** command.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify NTP is enabled by issuing the **show ntp** command.

Step 8 Using the following command, verify the GSS responds with the correct/valid response when sending a valid DNS A query to the name server in order to validate global sticky table entry: nslookup from both clients.

Step 9 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify the sticky entry was replicated to all GSS's, by issuing the **show sticky database all** command.

Step 10 Using the following commands, verify the GSS responds with the correct/valid response when sending valid DNS A queries to to the name server in order to validate the client receives the same answer in the sticky database from both GSS's: nslookup again from both clients

Step 11 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, inspect the sticky databases by issuing the **sticky database dump STICK format xml** and **type STICK** commands on all 4 GSS's, and validate.

Step 12 On the CSM in DCA, suspend the VIP's that resides in the sticky database on the GSS, by issuing the command **no inservice** for vserver wwwin-oefin, wwwin-oefin-9k, and wwwin-redirect. Verify the VIP/Answer is offline on all 4 GSS's by issuing the command **show statistics keepalive kalap list**.

Step 13 Issue the following command to verify the GSS responds with the correct/valid response when sending valid DNS A queries to dcap-gss-1.gslb.com in order to validate a new VIP is issued by the GSS, and the sticky database is updated: nslookup from client

Step 14 Verify the GSS responds with the correct/valid response when sending valid DNS A queries to the GSS in order to validate global sticky based on domain list. Verify the same answer is returned to the client by issuing the following commands: nslookup

Step 15 On the CSM in DCB, suspend the VIP's that resides in the sticky database on the GSS, by issuing the command **no inservice** for vserver wwwin-oefin, wwwin-oefin-9k, and wwwin-redirect. Verify the VIP/Answer is offline on all 4 GSS's by issuing the command **show statistics keepalive kalap list**.

Step 16 Verify the VIP/Answer is offline at DCB on all 4 GSS's by issuing the command **show statistics keepalive kalap list**.

Step 17 Verify the GSS responds with the correct/valid response when sending valid DNS A queries to the GSS in order to validate global sticky based on domain list. Verify the correct answer is returned to the client as both answers in clause #1 are down.

Step 18 Stop background scripts to collect final status of network devices and analyze for error.

Step 19 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the GSS to track the DNS response returned for a client D-proxy, and to return the same answer when the same client D-proxy makes a subsequent DNS request.
- We expect the GSS not to return an A record to a client for which the VIP on the GSS is deemed offline.
- We expect the GSS to replicate the DNS response to all 4 GSS's in the GSS network.

Results

Global Sticky Branch 1 and Branch 3 (Complete) passed.

Keepalives

This section contains the following topics:

GSS Kalap to CSM using VIP (Complete)

GSS kalap to CSM using VIP

Test Procedure

The procedure used to perform the GSS Kalap to CSM using VIP (Complete) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On both client machines, branch1-client-1 and branch3-client-1, ensure the primary and secondary name servers are 10.0.30.2 and 10.0.20.2, respectively.

Step 3 On both client machines, branch1-client-1 and branch3-client-1, flush the DNS resolver cache

Step 4 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com enable DNS console logging by issuing the command **logging disk enable** and the command **logging disk subsystem keepalive priority debugging**. Ensure keepalive logging is enabled by issuing the command **show logging**.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, clear the keepalive statistics by issuing the command **clear statistics keepalive all**.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, enable real-time logging by issuing the command **show log follow**.

Step 8 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify both CSM VIP's are online and reporting a load value of "2", by issuing the command **show statistics keepalive kalap list**.

Step 9 On the CSM at DCA "dca-agg-1" suspend the vserver "wwwin-oefin-9k" by issuing the command **no inservice** for the vserver "wwwin-oefin-9k".

Step 10 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify the GSS changed the load value to 85 for the VIP by looking at the real-time logging on each GSS.

Step 11 On the CSM at DCA "dca-agg-1" suspend the vserver "wwwin-oefin" by issuing the command **no inservice** for the vserver "wwwin-oefin".

Step 12 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify the GSS changed the load value to 170 for the VIP by looking at the real-time logging on each GSS and "show statistics keepalive kalap list"

Step 13 On the CSM at DCA "dca-agg-1" suspend the vserver "wwwin-redirect" by issuing the command **no inservice** for the vserver "wwwin-redirect".

Step 14 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify the GSS changed the load value to 255 for the VIP by looking at the real-time logging on each GSS and at the command, "show statistics keepalive kalap list".

Step 15 On both client machines, branch1-client-1 and branch3-client-1, preform an nslookup for the domain name "domain name" and verify you are receiving the correct answer back.

Step 16 Stop background scripts to collect final status of network devices and analyze for error.

Step 17 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect no CPU or memory problems.

Results

GSS Kalap to CSM using VIP (Complete) passed.

KAL-AP by TAG—Complete

GSS kalap to CSM using TAG

Test Procedure

The procedure used to perform the KAL-AP by TAG—complete test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On both client machines, branch1-client-1 and branch3-client-1, ensure the primary and secondary name servers are 10.0.30.2 and 10.0.10.2, respectively.

Step 3 On both client machines, branch1-client-1 and branch3-client-1, flush the DNS resolver cache

Step 4 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com enable DNS console logging by issuing the command **logging disk enable** and the command **logging disk subsystem keepalive priority debugging**. Ensure DNS logging is enabled by issuing the command **show logging**.

Step 6 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, clear the keepalive statistics by issuing the command **clear statistics keepalive all**.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, enable real-time logging by issuing the command **show log follow**. Verify the appropriate logging is displayed.

Step 8 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify all CSM VIP's are online and reporting a load value of "2", by issuing the command **show statistics keepalive kalap list**.

Step 9 On the CSM at DCA "dca-agg-1" remove the serverfarm "oracle-all" from the vserver "wwwin-oefin-9k" by issuing the command **no serverfarm oracle-all**

Step 10 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify the GSS changed the load value on the DCA VIP to a load value of "255" by looking at the real-time logging on each GSS and the "show statistics keepalive kalap list" command on the GSS.

Step 11 On the CSM at DCB "dcb-ss-1" remove the serverfarm "oracle-all" from the vserver "wwwin-oefin-9k" by issuing the command **no serverfarm oracle-all**

Step 12 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, verify the GSS changed the load value on both VIP's to a load value of "255" by issuing the command **show statistics keepalive kalap list** on all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect no CPU or memory problems.

Results

KAL-AP by TAG—complete passed.

LB Methods

This section contains the following topics:

LB Methods—Complete

This test verified that the GSS will respond to a well formed DNS query properly using the load balancing method defined in the DNS rule on the GSS.

Test Procedure

The procedure used to perform the LB Methods—complete test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On both client machines, branch1-client-1 and branch3-client-1, ensure the primary and secondary name servers are 10.0.10.2 and 10.0.30.2, respectively

Step 3 On both client machines, branch1-client-1 and branch3-client-1, flush the DNS resolver cache

Step 4 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com force a rotation and deletion of logs by issuing the command **rotate-logs**, and the command **rotate-logs delete-rotated-logs**.

Step 5 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, enable DNS console logging by issuing the **logging disk enable** and **logging disk subsystem dnsserver priority debugging** commands. Ensure DNS logging is enabled by issuing the **show logging** command.

Step 6 On the GSS, clear the DNS statistics by issuing the **clear statistics dns** command. Ensure the DNS statistics have been cleared by issuing the **show statistics dns global** command.

Step 7 On all 4 GSS's, dca-gss-1.gslb.dcap.com, dca-gss-2.gslb.dcap.com, dcb-gss-1.gslb.dcap.com, and dcb-gss-2.gslb.dcap.com, enable real time logging by issuing the **show log follow** command.

Step 8 On the GSS, configure the rule wwwin-oefin for round robin load balancing.

Step 9 Verify the GSS responds with the correct/valid response when sending DNS A queries for which the GSS is authoritative for by asking the GSS directly.

Step 10 On the GSS, configure the rule wwwin-oefin for weighted round robin load balancing.

Step 11 On the GSS, configure the answer wwwin-oefin-dca with a weight of 4 and verify the GSS responds with the correct/valid response when sending DNS A queries for which the GSS is authoritative.

Step 12 On the GSS, configure the rule wwwin-oefin with the answer group of "ordered_list_answers" along with a balance method of "roundrobin" for a return record count of eight and retest with nslookup, ask the GSS directly.

Step 13 On the GSS, configure the wwwin-oefin for ordered list load balancing.

Step 14 On the GSS, suspend the following answers: "order1, order2, and order3". Re-test with nslookup.

Step 15 On the GSS, configure the answer group ordered_list_answers in chronological order from one to eight, starting with 1.1.1.1 and ending with 8.8.8.8. Verify that the GSS responds with the correct/valid response in the correct order when sending DNS A queries for which the GSS is authoritative. Ask the GSS directly. From gss-linux-1 issue the following command:

Step 16 On the GSS, configure the wwwin-oefin rule for hashed load balancing and select hashed based on the domain name.

Step 17 Verify the GSS responds with the correct/valid response when sending DNS A queries for which the GSS is authoritative. Send requests to each of the four subdomains multiple times, in order to verify affinity for each subdomain. From gss-linux-1 issue the following commands:

```
dig @10.0.5.111 eng.gslb.com. a +qr
dig @10.0.5.111 hr.gslb.com. a +qr
dig @10.0.5.111 fin.gslb.com. a +qr
dig @10.0.5.111 market.gslb.com. a +qr
dig @10.0.5.102 eng.gslb.com. a +qr
dig @10.0.5.102 hr.gslb.com. a +qr
dig @10.0.5.102 fin.gslb.com. a +qr
dig @10.0.5.102 market.gslb.com. a +qrFrom gss-winxp-1 issue the following commands:
```

nslookup set d2 set type=a server 10.0.5.111 eng.gslb.com. hr.gslb.com. fin.gslb.com. market.gslb.com. server 10.0.5.102 eng.gslb.com. hr.gslb.com. fin.gslb.com. market.gslb.com.

Step 18 On the GSS, configure the wwwin-oefin rule for hashed load balancing and select hashed based on source address.

Step 19 Verify the GSS responds with the correct/valid response when sending DNS A queries for which the GSS is authoritative. Send requests to each of the four subdomains multiple times, to both name servers in order to verify affinity for each of the two name servers. Issue the following commands:

dig @10.0.5.111 eng.gslb.com. a +qr dig @10.0.5.111 hr.gslb.com. a +qr dig @10.0.5.111 fin.gslb.com. a +qr dig @10.0.5.111 market.gslb.com. a +qr dig @10.0.5.102 eng.gslb.com. a +qr dig @10.0.5.102 hr.gslb.com. a +qr dig @10.0.5.102 fin.gslb.com. a +qr dig @10.0.5.102 market.gslb.com. a +qr

Step 20 Stop background scripts to collect final status of network devices and analyze for error.

Step 21 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the correct load balancing method to be chosen and implemented by the GSS based on the load balancing algorithm defined in the DNS rule.

Results

LB Methods—complete passed.

CHAPTER 13

Bladeswitching

The HP c-Class BladeSystem is a complete infrastructure of servers, network management and storage, integrated in a modular design built to deliver the services vital to a business Data Center. The HP c-6700 enclosure provides all the power, cooling, and I/O infrastructure needed to support modular server, interconnect, and storage components. The enclosure is 10U high and holds up to 16 server and/or storage blades plus optional redundant network and storage interconnect modules. It includes a shared, 5 terabit per second, high-speed non-stop midplane for wire-once connectivity for server blades to network and shared storage. Power is delivered through a pooled-power backplane that ensures the full capacity of the power supplies is available to all server blades for maximum flexibility and redundancy.

The BladeSystem is ideal for large data centers, supporting up to 16 half-height, 2 or 4 socket Intel Xeon and or 2 socket AMD Opteron blades for maximum performance and density. By consolidating the modular components into one enclosure, power consumption can be reduced by up to 40% and airflow can be reduced by 47% compared to competitors' rack mount servers. Redundant and flexible I/O configurations, along with full power redundancy with N+N hot-plug power supplies and the flexibility of N+1 redundancy, make the system a highly available solution.

Consolidation also provides the ability to have simple to manage, easy to control, administration. With the Onboard Administrator, Integrated Lights Out (iLO), and HP Insight Control, you can manager your servers and Cisco switches by taking complete control regardless of the state of the server operating system or Cisco 3020 switch. The HP Insight Control Environment gives you the power to automate standard processes freeing up valuable IT resources. The pre-wired and pre-configured enclosure makes adding a new server blade as simple as plugging it in.

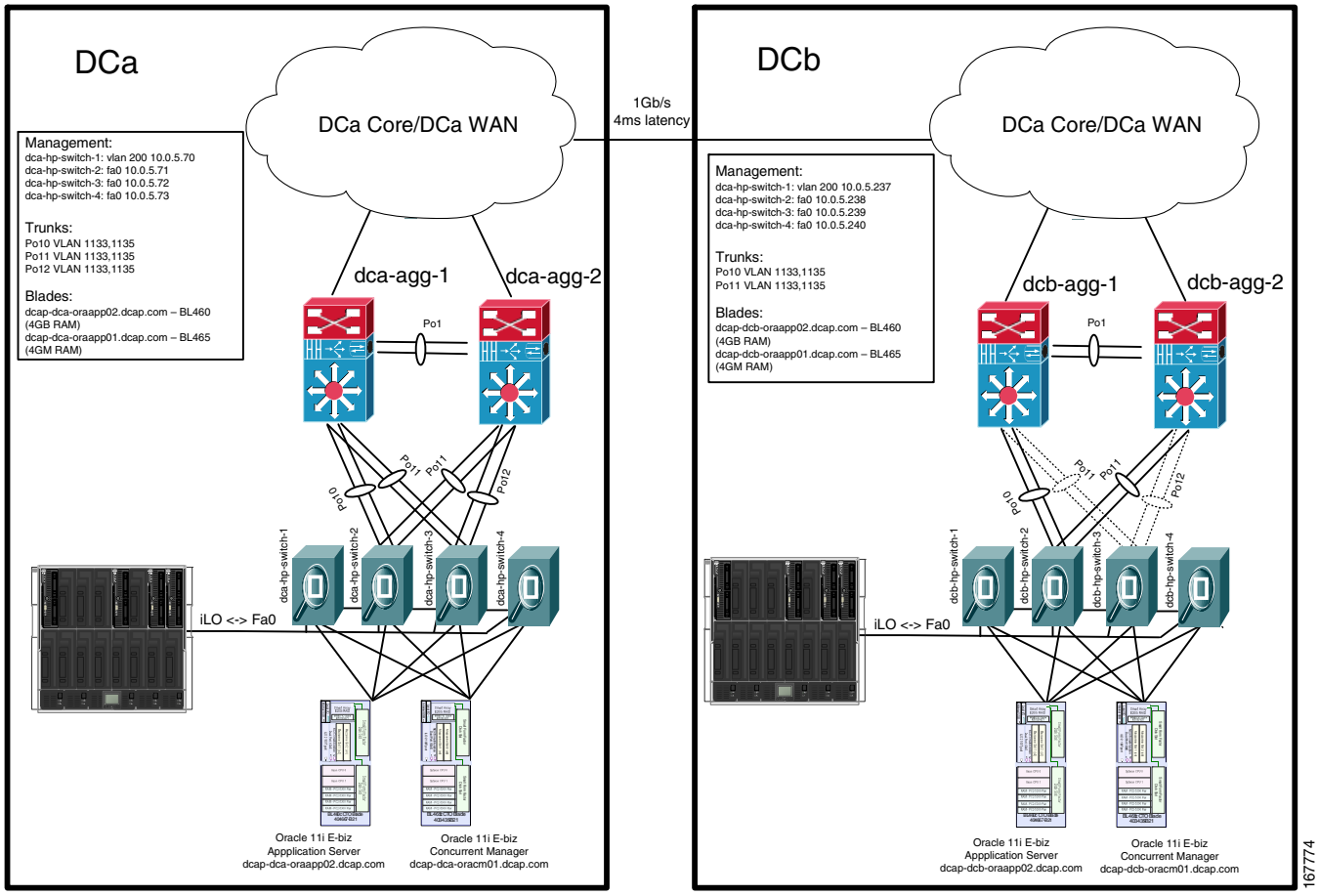
For network LAN connectivity, the Cisco Catalyst Blade Switch 3020 for HP is used. The 3020 is an integrated switch for HP c-Class BladeSystem customers that extends Cisco's resilient and secure Infrastructure Services to the server edge and utilizes existing network investments to help reduce operational expenses. The Cisco Catalyst Blade Switch 3020 for HP provides c-Class BladeSystem customers with an integrated switching solution which dramatically reduces cable complexity. This solution offers consistent network services like high availability, quality of service and security. It also utilizes Cisco's comprehensive management framework to simplify ongoing operations. Cisco's advanced network services, in combination with simplified management, helps reduce total cost of ownership.

Blader Servers Topology

In the DCAP topology both the Intel-based BL460c and AMD-based BL465c were provisioned to run the front end Oracle 11i E-Business Suite web application. BL685c servers were provisioned to provide back-end database service with Oracle Real Application Clusters (RAC). VMware ESX 3.0.2 was installed on BL485c servers, which were set up with boot from SAN and clustered to provide VMotioning capabilities. Each ESX server hosted Oracle Web application, Exchange Server 2003 hosts, and Windows Server 2003 domain controllers. The integrated Cisco 3020 Layer 2+ switch provided network connectivity to the data center Aggregation Layer in Data Center A. Four switches were housed in the DCA blade chassis and each one was configured with a dual-port Etherchannel dual homed to the Aggregation Layer switches. The Blade Enclosure in Data Center B was deployed with pass-thru modules allowing each server to connect directly into the Access Layer Catalyst 4948 and 6500 switches. The tests in this chapter focus on the basic feature functionality of the 3020 switch and its response to negative events.

Figure 13-1 shows the blade server topology configuration of the DCAP test topology.

Figure 13-1 HP Bladerservers Topology



Test Results Summary

Table 13-1 on page 13-3 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 13-1 on page 13-3 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTS's.

A number of resources were referenced during the design and testing phases of the HP Bladeservers in DCAP. These include the Data Center Blade Server Integration Guide, produced by Cisco's Enterprise Solution Engineering Data Center team. Links to this document is directly below. In Table 9-1, where applicable, pointers to relevant portions of this document are provided for reference purposes.

Data Center Blade Server Integration Guide (SRND):

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns304/c649/ccmigration_09186a00807ed7e1.pdf

Table 13-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Baseline, page 13-5	Baseline, page 13-5	1. Baseline Steady State 3020	
Baseline, page 13-5	CLI Functionality, page 13-6	1. PARSER RP via SSHv1 3020 2. PARSER RP via SSHv2 3020 3. PARSER RP via Telnet 3020	CSCsk62010 CSCsk65666 CSCsk65749
Baseline, page 13-5	Device Access, page 13-8	1. Repeated SSHv1 Logins 3020 2. Repeated SSHv2 Logins 3020 3. Repeated Telnet Logins 3020 4. VTY Access List 3020	
Baseline, page 13-5	Device Management, page 13-12	1. Local SPAN 3020 2. NTP Basic Functionality and Failover 3020 3. Remote SPAN 3020 4. SNMP MIB Walk 3020 5. SNMP Trap Functionality 3020 6. Syslog Basic Functionality 3020 7. Upgrade from Previously Certified Version	
Baseline, page 13-5	Security, page 13-18	1. Malformed SNMP Polling 3020 2. Malformed SSH Packets 3020 3. NMAP Open Port Scan 3020	
Layer 2 Protocols, page 13-21	Spanning Tree, page 13-22	1. RPVST+ Basic Functionality 3020	

Table 13-1 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Layer 2 Protocols, page 13-21	Trunking, page 13-23	1. 802.1q Basic Functionality 2. Layer 2 Trunk Failover	
Reliability, page 13-27	n/a	1. Power Cycle 3020	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Baseline

This encompasses all Baseline tests.

This section contains the following topics:

Baseline

Baseline tests verify network is in working order prior to starting testing and quantify steady state network performance.

This section contains the following topics:

Baseline Steady State 3020

This test verifies the network operation during steady state. While all background traffic and background routes are running, the network is allowed to run without perturbation to quantify the baseline CPU and memory of each device.

Test Procedure

The procedure used to perform the Baseline Steady State 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 While background traffic is running, allow the network to run in steady state for an extended period of time.

Step 3 Stop background scripts to collect final status of network devices and analyze for error.

Step 4 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that there will be no change in the test topology during the baseline period.
- We expect that there will be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Baseline Steady State 3020 passed.

CLI Functionality

Command Line testing robustly exercises the command line interface (CLI) of a router. The testing walks the parser tree, executing completed commands and filling in options as it comes to them. Certain branches of the parser tree were left out due to time constraints of the testing (eg. show tag-switching tdp, show mpls).

This section contains the following topics:

- PARSER RP via SSHv1 3020, page 13-6
- PARSER RP via SSHv2 3020, page 13-7
- PARSER RP via Telnet 3020, page 13-7

PARSER RP via SSHv1 3020

An automated script was used to test the valid **show** and **clear** commands on dca-hp-switch-4. SSH version 1 was used as the access protocol.

Test Procedure

The procedure used to perform the PARSER RP via SSHv1 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin executing the **show** and **clear** commands on the device under test.

Step 3 Stop background scripts to collect final status of network devices and analyze for error.

Step 4 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all commands to be executed without negative impact on the DUT.
- We expect no CPU or memory problems.

Results

PARSER RP via SSHv1 3020 failed. The following failures were noted: CSCsk62010, CSCsk65666, CSCsk65749.

PARSER RP via SSHv2 3020

An automated script was used to test the valid **show** and **clear** commands on dca-hp-switch-4. SSH version 1 was used as the access protocol.

Test Procedure

The procedure used to perform the PARSER RP via SSHv2 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin executing the **show** and **clear** commands on the device under test.

Step 3 Stop background scripts to collect final status of network devices and analyze for error.

Step 4 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all commands to be executed without negative impact on the DUT.
- We expect no CPU or memory problems.

Results

PARSER RP via SSHv2 3020 failed. The following failures were noted: CSCsk62010, CSCsk65666, CSCsk65749.

PARSER RP via Telnet 3020

An automated script was used to test the valid **show** and **clear** commands on dcb-hp-switch-4. Telnet was used as the CLI access protocol.

Test Procedure

The procedure used to perform the PARSER RP via Telnet 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin executing the **show** and **clear** commands on the device under test.

Step 3 Stop background scripts to collect final status of network devices and analyze for error.

Step 4 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all commands to be executed without negative impact on the DUT.
- We expect no CPU or memory problems.

Results

PARSER RP via Telnet 3020 failed. The following failures were noted: CSCsk62010, CSCsk65666, CSCsk65749.

Device Access

The Access test cases test the permissions to access the management port of the Catalyst 6500.

This section contains the following topics:

Repeated SSHv1 Logins 3020

Secure Shell (SSH) is an application and a protocol that provides secure replacement for the suite of Berkeley r-tools such as rsh, rlogin and rcp. (Cisco IOS supports rlogin.) The protocol secures the sessions using standard cryptographic mechanisms, and the application can be used similarly to the Berkeley rexec and rsh tools.

This test verified that repeated SSHv1 logins to a Cisco 3020 switch did not impact memory or system stability. The device dcb-hp-switch-4 was subjected to 1000 login attempts, using version 1 of the SSH protocol, by six concurrent iterations of a login script. This was done to max out the vty lines on the device. It was verifed that all logins were successful and that system performance was not affected.

Test Procedure

The procedure used to perform the Repeated SSHv1 Logins 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the HTTP background traffic is running by issuing the **show interface port-channel *channel-id*** counters command.

Step 3 Verify that dca-hp-switch-4 is configured for ssh login using the **show ip ssh** command.

The **show ip ssh** command should show **SSH Enabled—version 1.99** in the output.

Step 4 Initiate 6 iterations of the test script. Each iteration will attempt to log into the switch 1000 times, successively, using SSH version 1. Upon successful login the script will issue the **show version** and **show process memory** commands.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that there will be no system error messages resulting from the multiple, repeated SSH login attempts.
- We expect no CPU or memory problems.

Results

Repeated SSHv1 Logins 3020 passed.

Repeated SSHv2 Logins 3020

Secure Shell (SSH) is an application and a protocol that provides secure replacement for the suite of Berkeley r-tools such as rsh, rlogin and rcp. (Cisco IOS supports rlogin.) The protocol secures the sessions using standard cryptographic mechanisms, and the application can be used similarly to the Berkeley rexec and rsh tools.

This test verified that repeated SSHv1 logins to a Cisco 3020 switch did not impact memory or system stability. The device dcb-hp-switch-4 was subjected to 1000 login attempts, using version 2 of the SSH protocol, by six concurrent iterations of a login script. This was done to max out the vty lines on the device. It was verifed that all logins were successful and that system performance was not affected.

Test Procedure

The procedure used to perform the Repeated SSHv2 Logins 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the HTTP background traffic is running.

Step 3 Verify that dca-hp-switch-4 is configured for ssh login using the **show ip ssh** command.

The **show ip ssh** command should show **SSH Enabled—version 1.99** in the output.

Step 4 Initiate 6 iterations of the test script. Each iteration will attempt to log into the switch 1000 times, successively, using SSH version 2. Upon successful login the script will issue the **show version** and **show process memory** commands.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that there will be no system error messages resulting from the multiple, repeated SSH login attempts.
- We expect no CPU or memory problems.

Results

Repeated SSHv2 Logins 3020 passed.

Repeated Telnet Logins 3020

This test verified that repeated Telnet logins to a Cisco 3020 switch did not impact memory or system stability. The device dca-hp-switch-4 was subjected to 1000 telnet login attempts by six concurrent iterations of a login script. This was done to max out the VTY lines on the device. It was verified that all logins were successful and that system performance was not affected.

Test Procedure

The procedure used to perform the Repeated Telnet Logins 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the HTTP background traffic is running.

Step 3 Initiate 6 iterations of the test script. Each iteration will attempt to log into the switch 1000 times, successively, using Telnet. Upon successful login the script will issue the **show version** and **show process memory** commands.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that there will be no system error messages resulting from the multiple, repeated Telnet login attempts.
- We expect no CPU or memory problems.

Results

Repeated Telnet Logins 3020 passed.

VTY Access List 3020

The **access-class** command is used to restrict inbound or outbound telnet access for VTY sessions. Only numbered access lists can be applied to VTY lines.

This test verified the operation of IP access-class lists on 3020. Multiple hosts (lickskillet and buladean) were denied, then permitted, access to the DUT by applying different access-class lists to its VTY lines.

Test Procedure

The procedure used to perform the VTY Access List 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Access the DUT through the console and verify the configuration of access lists one and two using the **show access-lists** command.

Access list one should permit IP address on the 172.18.179.128/27 subnet. Access list two should permit only the IP address, 172.18.177.129.

Step 3 Verify that telnet access is permitted to the DUT from the lickskillet server which has the IP address, 172.18.179.140.

Step 4 Restrict access to those hosts defined in access list one(172.18.179.128/27) on the VTY lines of the DUT using the **line vty** and **access-class 1 in** commands.

Step 5 Verify that access is not permitted from the hosts within the subnet defined by access list one by attempting to telnet to the DUT from a device within the subnet.

You should see a connection refused message.

Step 6 Use the **no access-class 1 in** command to remove the access list from the VTY lines.

Step 7 Verify that telnet access is again permitted from lickskillet.

Step 8 Restrict access to the host defined in access list two on the VTY lines by issuing the **access-class 2 in** command.

Step 9 Verify that telnet access is permitted from the host specified in the access list two.

Step 10 Verify that telnet access is not permitted from hosts not specified in the access list two.

Step 11 Issue the **no access-class 2 in** command to remove the access list from the VTY lines.

Step 12 Verify that telnet access is once again permitted.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the IP access-class list to appropriately allow or deny access to the connecting host.
- We expect that there will be no sustained or unexpected impact on either the CPU or memory during this test.

Results

VTY Access List 3020 passed.

Device Management

Simple Network Management Protocol (SNMP) is an application-layer protocol designed to facilitate the exchange of management information between network devices.

This section contains the following topics:

Local SPAN 3020

Local SPAN selects network traffic to send to a network analyzer. SPAN should not affect the switching of network traffic on source ports or VLAN's. SPAN sends a copy of the packets received or transmitted by the source ports and VLAN's to a destination port dedicated for SPAN use.

This test verified that normal traffic forwarding was maintained when a local SPAN session was configured on dca-hp-switch-4. Interface Port-channel 11 was used as the SPAN source. The SPAN destination was a local port, GigabitEthernet0/21. The network was monitored for traffic irregularities and the DUT was monitored for CPU or memory stability.

Test Procedure

The procedure used to perform the Local SPAN 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the switch use the **show monitor** command to verify that there are no SPAN sessions present.

Step 3 Configure the SPAN source to be interface Port-channel 11 using the **monitor session 1 source interface Port-channel 11 both** command. By specifying both, the session will SPAN ingress and egress traffic on the port-channel.

Step 4 Configure the SPAN destination to be interface Gi0/21 using the **monitor session 1 destination interface Gi0/21** command.

Step 5 Clear the traffic counters on the switch using the **clear counters** command.

Step 6 Run background HTTP test traffic for a period of 5 minutes. The HTTP server is simulated by a test port connected to Gi0/18 of the switch. The HTTP client is making HTTP get requests of 200k files from a host in DCB.

Step 7 Compare the counters of the SPAN source interface with those of the SPAN destination interface using the **show interface***interface***counters** command.

The SPAN source is monitoring both transmit and receive of the source interface. The SPAN destination interface egress counters should reflect the combination of both directions of traffic on the SPAN source.

Step 8 Look for any errors on the SPAN destination interface using the **show interfaces GigabitEthernet0/21** command.

Step 9 Remove the SPAN configuration from the switch using the **no monitor session 1** command.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the SPAN utility will operate soundly under load.
- We expect that the SPAN utility will not interfere with normal network traffic.
- We expect that there will be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Local SPAN 3020 passed.

NTP Basic Functionality and Failover 3020

Network Time Protocol (NTP) synchronizes timekeeping among a set of distributed time servers and clients. This synchronization allows events to be correlated when system logs are created and other time-specific events occur. An NTP server must be accessible by the client switch. NTP runs over User Datagram Protocol (UDP), which runs over IP.

This test verified the basic functionality of NTP on the Cisco 3020. A local Sun server (IP address: 172.18.177.150) and a local 6500 running native were used as NTP servers. A cisco 3020 was configured as the NTP client. There are two NTP servers configured on each device in the test network. This test also verified the ability of the device under test to switchover to the backup NTP server in the case of a primary server failure.

Test Procedure

The procedure used to perform the NTP Basic Functionality and Failover 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the servers with IP addresses 172.18.177.131 and 172.18.177.150 (preferred) are configured as the NTP servers on the DUT and that 172.18.177.150 is synced to it as the master.

The two servers are configured on dca-hp-switch-4. The server 172.18.177.131 is (+) selected, as it is a qualified NTP server. The server 172.18.177.150 is (*) master (synced).

Step 3 On the server 172.18.177.150, stop the NTP daemon.

Step 4 Verify that the server 172.18.177.131 is now the master server and that NTP is functioning again.

On the DUT, both servers are (~) configured. The server 172.18.177.131 is now (*) master (synced), while the server 172.18.177.150 is not even (+) selected. The device is synchronized to the new reference server, 172.18.177.131.

NOTE: Synchronization to the new server can take over 2 hours as the timeout interval is long.

Step 5 Start the NTP daemon on the server 172.18.177.150 again, and verify that it once again becomes the active master server for the DUT.

Verify the device once again has the original NTP status, with 172.18.177.150 as the NTP server-of-choice.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the test device is synchronized to the configured NTP server, and that the clocks on the device and the server are in sync.
- We expect no CPU or memory problems.

Results

NTP Basic Functionality and Failover 3020 passed.

Remote SPAN 3020

With remote SPAN, the SPAN destination is a VLAN, rather than a physical interface. This VLAN is configured as a remote VLAN throughout the network. Traffic that is copied to the SPAN VLAN is tagged with that VLAN ID and sent through the network to a traffic analyzer attached to a network device that is remote to the SPAN source.

This test verified that normal traffic forwarding was maintained when a remote SPAN session was configured on dca-hp-switch-4. Interface Port-channel 11 was used as the SPAN source. The SPAN destination was VLAN1140. The network was monitored for traffic irregularities and the DUT was monitored for CPU or memory stability.

Test Procedure

The procedure used to perform the Remote SPAN 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 On the switch use the **show monitor** command to verify that there are no SPAN sessions present.

Step 3 Configure the SPAN source to be interface Port-channel 11 using the**monitor session 1 source interface Port-channel 11 both** command. By specifying both, the session will SPAN ingress and egress traffic on the Portchannel.

Step 4 Configure the SPAN destination to be remote VLAN 900 using the **monitor session 1 destination remote VLAN 900** command.

Step 5 Verify that interface GigabitEthernet 0/21 is configured to trunk VLAN 900 with the **show interface GigabitEthernet0/21 trunk** command.

Step 6 Clear the traffic counters on the switch using the **clear counters** command.

Step 7 Run background HTTP test traffic for a period of 5 minutes. The HTTP server is simulated by a test port connected to Gi0/18 of the switch. The HTTP client is making HTTP get requests of 200k files from a host in DCB.

Step 8 Compare the counters of the SPAN source interface with those of the SPAN destination VLAN using the **show interface***interface***counters** command.

The SPAN source is monitoring both transmit and receive of the source interface. The SPAN destination interface egress counters should reflect the combination of both directions of traffic on the SPAN source.

Step 9 Look for any errors on the SPAN destination interface using the **show interfaces GigabitEthernet 0/21** command.

Step 10 Remove the SPAN configuration from the switch using the **no monitor session 1** command.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the remote SPAN utility will operate soundly under load.
- We expect that the remote SPAN utility will not interfere with normal network traffic.
- We expect that there will be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Remote SPAN 3020 passed.

SNMP MIB Walk 3020

Simple Network Management Protocol (SNMP) is ubiquitous as a tool used by network administrators to manage network performance, find and solve network problems, and plan for network growth.

This test verified that a SNMP walk of the MIB tree of dca-hp-switch-2 did not cause any memory loss, tracebacks, or crashes. From a server, six hours worth of version 1 SNMP walks were performed.

Test Procedure

The procedure used to perform the SNMP MIB Walk 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the SNMP configuration of dca-hp-switch-2 using the **show running-config** command.

Step 3 From the server CLI perform one thousand SNMP walks on the DUT using the **snmpwalk** utility.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that no tracebacks or crashes will occur on the DUT.
- We also expect no memory loss will occur.

Results

SNMP MIB Walk 3020 passed.

SNMP Trap Functionality 3020

SNMP traps enable an agent to notify the management station of significant events by way of an unsolicited SNMP message.

This test verified the basic SNMP trap functionality of the 3020. In this procedure, an SNMP trap is created by entering and leaving config mode on dca-hp-switch-4. The logging messages created on the device should be logged to the SNMP trap receiver.

Test Procedure

The procedure used to perform the SNMP Trap Functionality 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that dca-hp-switch-4 is configured to send SNMP traps generated by configuration messages to the server 172.18.177.140 by issuing the **show running-config** command.

Step 3 Verify connectivity with the server 172.18.177.140 and configure configuration traps on the DUT.

Step 4 Configure the server(172.18.177.140) to accept the traps.

Step 5 Enter and leave configuration mode by issuing **configure terminal** and **end** commands, generating a log message.

Step 6 Verify that the traps are received by a machine that is set up as the SNMP trap receiver. View the output from the log files of that machine.

Step 7 Stop the trap daemon on the server and unconfigure the DUT by issuing the **no snmp-server enable traps config** command.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that SNMP functions according to specifications, generating and sending a trap to the configured host.
- We expect no CPU or memory problems.

Results

SNMP Trap Functionality 3020 passed.

Syslog Basic Functionality 3020

The System Log (syslog) protocol provides a transport to allow a machine to send event notification messages across IP networks to event message collectors, which are also known as syslog servers.

This test verified the basic functionality the Syslog feature on the Cisco 3020. A syslog host is defined and a syslog message is generated on the switch. It is verified that the log message is also seen on the syslog server host.

Test Procedure

The procedure used to perform the Syslog Basic Functionality 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that syslog is configured on dca-hp-switch-4 and that the designated server is 172.18.177.150 by issuing the **show running-config** command.

The DUT is configured to send logging messages to server 172.18.177.132.

Step 3 On the DUT and enter enable mode and perform the **shutdown** and **no shutdown** commands on port-channel 11.

Step 4 Display output from the syslog server and compare it to messages received on the DUT. The syslog server logged the debug messages from dca-hp-switch-4 (10.0.4.227).

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect each message that is logged to the VTY session will also be logged to the syslog server.
- We expect no CPU or memory problems.

Results

Syslog Basic Functionality 3020 passed.

Upgrade from Previously Certified Version

This test verified the ability for the Cisco 3020 to be upgraded from the previously certified version of code to the latest available version. The access layer device, dca-hp-switch-2, was upgraded to ensure that all hardware and configurations at the access layer were upgraded without issue.

Test Procedure

The procedure used to perform the Upgrade from Previously Certified Version test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the DUT is running the old Native Cisco IOS image using the **show version** command.

Step 3 Issue the **archive download-sw /overwrite /reload tftp://172.18.177.150/cbs30x0-lanbasek9-tar.122-35.SE.tar** command to download, extract, and install the new image to flash. Because of the /reload argument the switch will be reloaded after successful install. Monitor the switch via the console so the reload is logged.

Step 4 Once the switch has rebooted, issue the **show version** commands to verify that the DUT came back online successfully and that the new image is running without errors.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the upgrade process 3020 platform will proceed smoothly and without error.
- We expect no CPU or memory problems.

Results

Upgrade from Previously Certified Version passed.

Security

Secure Shell (SSH) is an application and a protocol that provide a secure replacement to the Berkeley r-tools. The protocol secures the sessions using standard cryptographic mechanisms, and the application can be used similarly to the Berkeley rexec and rsh tools. There are currently two versions of SSH available: SSH Version 1 and SSH Version 2.

The SSH Server feature enables a SSH client to make a secure, encrypted connection to a Cisco router. This connection provides functionality that is similar to that of an inbound Telnet connection. Before SSH, security was limited to Telnet security. SSH allows a strong encryption to be used with the Cisco IOS software authentication. The SSH server in Cisco IOS software will work with publicly and commercially available SSH clients.

The user authentication mechanisms supported for SSH are RADIUS, TACACS+ and the use of locally stored user names and passwords.

This section contains the following topics:

Malformed SNMP Polling 3020

Each network device in the Data Center test topology is configured for both read-only and read-write access via SNMP. The availability of SNMP access of certain network devices to the outside world leaves them vulnerable to certain attacks. One possible attack is through the use of malformed SNMP packets.

This test relies on the Protos (http://www.ee.oulu.fi/research/ouspg/protos/) test suite for SNMP. This test application subjects the DUT to many hundreds of misconfigured SNMP packets in an attempt to disrupt system activity. The Protos SNMP test was run against device dca-hp-switch-4 while that device was being monitored for errors and disruptions to CPU and memory stability.

Test Procedure

The procedure used to perform the Malformed SNMP Polling 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 If the background test traffic is not already running, start it now.

Step 3 Verify the SNMP community string settings default using the **show running-config** command on dca-hp-switch-4.

The read-only password is public (default).

Step 4 Execute the two Protos traffic generation scripts on dca-hp-switch-4.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all DUT's not to pause indefinitely, crash, or give any tracebacks while test is being run.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Malformed SNMP Polling 3020 passed.

Malformed SSH Packets 3020

Similar to its vulnerability to outside attacks via corrupt SNMP traffic, a network device may be susceptible to outside attacks via corrupt SSH traffic. This test relies on the Protos (http://www.ee.oulu.fi/research/ouspg/protos/) test suite for SSH. This test application subjects the DUT to many hundreds of misconfigured SSH packets in an attempt to disrupt system activity.

The Protos SSH test was run against the data center test network device dca-hp-switch-4 while that device was being monitored for errors and disruptions to CPU and memory stability.

Test Procedure

The procedure used to perform the Malformed SSH Packets 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 If the background test traffic is not already running, start it now.

Step 3 Verify that dca-hp-switch-4 is configured with a hostname, domain name, and TACACS authentication on the VTY lines using the following commands:

- show running-config | include hostname|domain|aaa|tacacs
- show running-config | begin line vty 0

The lines that should be present are as follows:

```
hostname dca-hp-switch-4
aaa new-model
aaa authentication login default group tacacs+ local
aaa authorization exec default group tacacs+ if-authenticated local
aaa session-id common
ip domain-name example.com
tacacs-server host 172.18.177.150
tacacs-server directed-request
tacacs-server key cisco
line vty 0 4
 transport input telnet ssh
```

Step 4 Verify the SSH server on dca-hp-switch-4 is enabled using the **show ip ssh** command and that it is accepting SSH connections.

Step 5 Send malformed SSH packets to the device while monitoring the device. Ensure that the device does not pause indefinitely, crash, or reload.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect SSH vulnerability testing not to cause the switch to reload, pause indefinitely, or crash.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

Malformed SSH Packets 3020 passed.

NMAP Open Port Scan 3020

A common way for hackers to wreak havoc on a network is to scan a network device (or an endpoint) for open TCP or UDP ports using the freely available NMAP tool. If an open port is found, the hacker may be able to exploit it and disrupt system activity. It is important, therefore, that a network device leave only those ports open that need to be for normal network services.

The test devices in the Data Center test topology have certain ports open by design. These include Telnet (port 23), SSH (22), and HTTPS (443). This test runs the NMAP Port scan tool against each device in the test topology, verifying that no ports open other than the ones expected. The DUT's are monitored for errors and CPU and memory stability during this procedure.

Test Procedure

The procedure used to perform the NMAP Open Port Scan 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Begin a port span on the 3020 devices in the test bed using the NMAP tool.

The command, run as root, that was used to execute this step was **nmap -v -p 1-65535***target_ip*.

Step 3 Verify that all open ports (as revealed by the port scan) are expected.

Each of the devices in the data center blade test topology have Telnet (TCP port 23), SSH (TCP 22), and HTTPS (443) open. These are the only ports we expect to see open. TCP Port 49623 is open when using Fast Ethernet 0 for management so it is expected that this will be seen on switches set up with iLO sourced management.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the open ports to be revealed by the NMAP tool.
- We expect Telnet (TCP port 23), SSH (TCP port 22), HTTPS (443) to be open.
- On switches using the Fa0 port for iLO management we expect port TCP port 49623.
- We expect there to be no sustained or unexpected impact on either the CPU or memory during this test.

Results

NMAP Open Port Scan 3020 passed.

Layer 2 Protocols

This encompasses layer 2 of the protocol stack.

This section contains the following topics:

Spanning Tree

The IEEE 802.1d Spanning Tree specification allows physical path redundancy without active network "loops" by defining a tree that spans all of the switches in an extended network and then forces certain redundant data paths into a standby (blocked) state. At regular intervals, the switches in the network send and receive spanning tree packets that they use to identify the path. If one network segment becomes unreachable, or if spanning tree costs change, the spanning tree algorithm reconfigures the spanning tree topology and reestablishes the link by activating the standby path.

This section contains the following topics:

RPVST+ Basic Functionality 3020

The default spanning-tree configuration for all swithces under test is Rapid-PVST+. This spanning-tree mode is the same as PVST+ except that is uses a rapid convergence based on the IEEE 802.1w standard. To provide rapid convergence, the rapid PVST+ immediately deletes dynamically learned MAC address entries on a per-port basis upon receiving a topology change. By contrast, PVST+ uses a short aging time for dynamically learned MAC address entries.

The rapid PVST+ uses the same configuration as PVST+ (except where noted), and the switch needs only minimal extra configuration. The benefit of rapid PVST+ is that you can migrate a large PVST+ install base to rapid PVST+ without having to learn the complexities of the MSTP configuration and without having to reprovision your network. In rapid-PVST+ mode, each VLAN runs its own spanning-tree instance up to the maximum supported.

This test verified the basic functionality of rPVST+ on the Cisco 3020. In the standard configuration a Portchannel is trunked to both aggregation switches. One of the Portchannels goes into a blocking state to maintain a loop free environment. The test verified the ability of the switch to go from Blocking to Forwarding upon a Port-channel failure, as well as, the ability for the switch to go back to its normal Blocking/Forwarding state once that Portchannel has been restored.

Test Procedure

The procedure used to perform the RPVST+ Basic Functionality 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the switch is running in rPVST+ mode by issuing the **show spanning-tree summary** command.

Step 3 Verify the STP state is Forwarding for Portchannel 11 and Blocking for Portchannel 12 for VLAN 1133 on the switch by issuing the **show spanning-tree vlan 1133** and **show spanning-tree interface port-channel *port-channel*** commands.

Step 4 Shutdown Portchannel 10 on the switch by issuing the **shutdown** command.

Step 5 Verify the STP state for Portchannel 12 has transitioned from Blocking to Forwarding by issuing the **show spanning-tree vlan 1133** and **show spanning-tree interface port-channel *port-channel*** commands.

Step 6 Bring up Portchannel 11 by issuing the **no shutdown** command.

Step 7 Verify the STP state for Portchannel 12 has transitioned from Forwarding to Blocking and that the STP state for Portchannel 11 has returned to Forwarding by issuing the **show spanning-tree vlan 1133** and **show spanning-tree interface port-channel *port-channel*** commands.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the switch will be operating in Rapid-PVST+ mode.
- We expect the switch will be Forwarding on VLAN 1133 to one aggregation switch and Blocking on another.
- We expect that the switch will begin Forwarding on the Portchannel that was in the Blocking state once the Forwarding Portchannel is brought down.
- We expect the switch to transition back to its normal Forwarding/Blocking state once the Portchannel is brought back up.
- We expect no CPU or memory problems.

Results

RPVST+ Basic Functionality 3020 passed.

Trunking

A trunk is a point-to-point link between one or more switch ports and another networking device such as a router or a switch. Trunks carry the traffic of multiple VLANs over a single link and allow VLANs to be extended across an entire network. The table lists and describes the five modes of trunking on Cisco switches.

Mode	Description
On	Local interface trunks. Sends Dynamic Trunking Protocol (DTP) packets. Puts the port into permanent trunking mode and negotiates to convert the link to a trunk link. The port becomes a trunk port even if the neighboring port does not agree to the change.
Off	Local interface does not trunk. Puts the port into nontrunking mode and negotiates to convert the link into a nontrunk link. The port becomes a nontrunk port even if the neighboring port does not agree to the change.
Auto	Local interface trunks if it receives DTP packets. Enables the port to convert the link to a trunk link. The port becomes a trunk port if the neighboring port is set to on or desirable mode. This is the default mode for Fast Ethernet and Gigabit Ethernet ports.
Desireable	Local interface sends DTP packets. Makes the port actively attempt to convert the link to a trunk line. The port becomes a trunk port if the neighboring port is set to on, desirable, or auto mode.

Mode	Description
Nonegotiate	Local interface forms a trunk and does not send DTP packets. Puts the port into permanent trunking mode, but prevents the port from generating DTP frames. You must configure the neighboring port normally as a trunk port to establish a trunk link.

This section contains the following topics:

802.1q Basic Functionality

On Cisco 3020 swithces trunks can be formed in multiple ways. Trunking can either be dynamic, in which trunking is negotiated between the two sides of the link, or it can be statistically set to on or off. In the case of the Data Center test topology, the trunk links are set to on, meaning that they will trunk VLAN's regardless of what the remote side of the link is doing.

The trunk encapsulation can also be either dynamically negotiated or set statically. In the Data Center test topology, the encapsulation is set statically to 802.1q, or dot1q.

This test verified that the links that were configured as trunk links between the Data Center devices actually formed trunks correctly. The links looked at include those between a Cisco 3020 and a Catalyst 6500(dca-hp-switch-3 and dca-agg-1). The CPU and memory utilization of the DUT's was monitored for stability.

Test Procedure

The procedure used to perform the 802.1q Basic Functionality test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 The devices dca-agg-1, a Catalyst 6500, and dca-hp-blade-4, a Cisco 3020, are connected by a static trunk. Use the **show running-config interface***interface* and **show interfaces***interface***trunk** commands to verify that this is the current configuration and the trunk is currently working.

Step 3 If not already running, start the background HTTP test traffic. The HTTP server is simulated by a test port connected to Gi0/18 of the switch. The HTTP client is making HTTP get requests of 200k files from a host in DCB.

Step 4 Verify on both devices that traffic is passing without error by issuing the **show interface port-channel *port-channel*** and **show interface port-channel *port-channel*** counters commands.

From the output verify that no drops are occurring and that traffic is passing across bidirectionally on the Portchannel.

Step 5 Using the **shutdown** and **no shutdown** commands, flap the Portchannel interface on dca-hp-switch-4.

Step 6 Use the **show interfaces***interface***trunk** command to verify that the trunk between dca-agg-1 and dca-hp-switch-4 has re-formed correctly.

Step 7 Verify on both devices that traffic is once again passing without error by issuing the **show interface port-channel *port-channel*** and **show interface port-channel *port-channel*** counters commands.

From the output verify that no drops are occurring and that traffic is passing across bidirectionally on the Portchannel.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the 802.1q trunks to be formed correctly between the Cisco 3020 and the Catalyst 6500.
- We expect traffic to pass successfully over the trunk.
- We expect the trunk to reform after a failure.
- We expect the trunk will once again pass traffic once it has reformed.
- We expect no CPU or memory problems.

Results

802.1q Basic Functionality passed.

Layer 2 Trunk Failover

Layer 2 trunk failover, also known as link-state tracking, is a feature that provides Layer 2 redundancy in the network when used with server NIC adapter teaming. When the server network adapters are configured in a primary or secondary relationship known as teaming, if the link is lost on the primary interface, connectivity is transparently switched to the secondary interface. When you enable Layer 2 trunk failover on the switch, the link state of the internal downstream ports are bound to the link state of one or more of the external upstream ports. An internal downstream port is an interface that is connected to the server. An external upstream port is an interface that is connected to the external network. When you associate a set of downstream ports to a set of upstream ports, if all of the upstream ports become unavailable, trunk failover automatically puts all of the associated downstream ports in an error-disabled state. This causes the server primary interface to failover to the secondary interface. When Layer 2 trunk failover is not enabled, if the upstream interfaces lose connectivity, (the external switch or router goes down, the cables are disconnected, or link is lost), the link state of the downstream interfaces remain unchanged. The server is not aware that external connectivity has been lost and does not failover to the secondary interface. An interface can be an aggregation of ports (an EtherChannel), or a single physical port in access or trunk mode. Each downstream interface can be associated with one or more upstream interfaces. Upstream interfaces can be bundled together, and each downstream interface can be associated with a single group consisting of multiple upstream interfaces. These groups are referred to as link-state groups. In a link-state group, the link states of the downstream interfaces are dependent on the link states of the upstream interfaces. If all of the upstream interfaces in a link-state group are in the link-down state, the associated downstream interfaces are forced into the link-down state. If any one of the upstream interfaces in the link-state group is in a link-up state, the associated downstream interfaces can change to or remain in the link-up state.

Follow these guidelines to avoid configuration problems:

- Do not configure a cross-connect interface (gi0/23 or gi0/24) as a member of a link-state group.
- Do not configure an EtherChannel as a downstream interface.
- Only interfaces gi0/1 through gi0/16 can be configured as downstream ports in a specific link-state group.

- Only interfaces gi0/17 through gi0/24 can be configured as upstream ports in a specific link-state group.
- An interface that is defined as an upstream interface cannot also be defined as a downstream interface in the same or a different link-state group.
- The reverse is also true.
- An interface cannot be a member of more than one link-state group.
- You can configure only two link-state groups per switch.

Test Procedure

The procedure used to perform the Layer 2 Trunk Failover test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the switch is configured globally to track Layer 2 link state with the **link state track** ***[1-2]*** commands.

Step 3 Verify the upstream Portchannel interfaces are configured with the **link state group** ***[1-2]*** upstream command.

Step 4 Verify one of the downstream interfaces per link state group is configured with the **link state group** ***[1-2]*** downstream command.

Step 5 Verify the status of the upstream portchannel and downstream port with the **show interface** ***interface*** status and **show link state group detail** commands.

Step 6 Shutdown the upstream Portchannel 11 and portchannel 12 with the **shutdown** command.

Step 7 Verify the downstream port configured as a part of link state group 1's status has been changed to err-disabled by issuing the **show interface** ***interface*** status and **show link state group detail** commands.

Step 8 Bring back up the upstream Portchannels,configured for upstream link state group 1 and 2, with the **no shutdown** command.

Step 9 Verify the downstream ports status has returned to connected by issuing the **show interface** ***interface*** status and **show link state group detail** commands.

Step 10 Unconifgure link state tracking for group 2 with the **no link state track 2** command.

Step 11 Shutdown the upstream Portchannel 11 and Portchannel 12 with the **shutdown** command.

Step 12 Verify that only ports set up to track link state for group 1 were moved into the err-disable state with the **show link state group detail** command.

Step 13 Reconfigure link state tracking globally for group 2 with the **link state track 2** command.

Step 14 Verify that the downstream port configured for link state group 2 moves to the err-disable state by issuing the **show link state group detail** command.

Step 15 Bring back up both Portchannels with the **no shutdown** commands and verify the ports in the corresponding link state groups return to Interface up status by issuing the **show link state group detail** command.

Step 16 Stop background scripts to collect final status of network devices and analyze for error.

Step 17 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the state of the downstream port will go to err-disable state when the upstream interface is brought down.
- We expect the state of the downstream port to go back to an UP state once the upstream interface is brought back up.
- We expect no CPU or memory problems.

Results

Layer 2 Trunk Failover passed.

Reliability

Reliability of network devices is essential for keeping the network functional. WAEs reliability testing included reloading the devices and verifying the configuration was restored after boot up.

This section contains the following topics:

Power Cycle 3020

This test verified the ability of the Cisco 3020 to recover from a power failure. Through the HP onboard administrator GUI a power failure was simulated and it was verified that the switch booted and resumed normal working status after the failure.

Test Procedure

The procedure used to perform the Power Cycle 3020 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 While monitoring the console of dca-hp-switch-4, simulate a power failure in the HP Onboard administrator GUI. In the GUI select the Cisco Catalyst Blade Switch in Bay 4 and click the virtual buttons tab. On this page select the Reset button to force a reset of the switch.

Step 3 Stop background scripts to collect final status of network devices and analyze for error.

Step 4 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect the device to boot and come back online without error.
- We expect that no tracebacks or crashes will occur on the DUT.
- We also expect no memory loss will occur.

Results

Power Cycle 3020 passed.

CHAPTER 14

Oracle 11i E-Business Suite

Oracle E-Business Suite is a fully integrated, comprehensive suite of enterprise business applications that provide quality business information for effective decision-making. It allows adaptation that lends optimal responsiveness, offering best practices with industry-specific capabilities necessary to augment competitive change. Oracle 11i dramatically lowers IT and business costs by improving business processes, reducing customizations, decreasing integration costs, and consolidating instances.

The centerpiece of the DCAP topology, with respect to Oracle application testing, is configuration of Oracle 11i E-Business Suite 11.5.10.2 with Oracle 10gR2 RAC in Active/Active Hybrid mode implemented across two active data centers. The Application Tier is shared across two data centers making it Active/Active while the Database Tier is active in only one data center (DCa) with data replicating synchronously to the second Data Center (DCb), making it Active/Passive. This architecture, as deployed, meets the functional requirements for Oracle 11i as well as providing a solution for enterprises that offers business resilience, high availability, scalability and security. The Oracle Vision demo environment is leveraged for application testing which includes running real application traffic using industry-standard Mercury Load Runner tool. Traffic is sent to both data centers, DCa and DCb, with WAAS and without WAAS from clients located at three branch offices.

Note Failover and failback testing was conducted against this implementation of the Oracle application. Refer to Volume 11: Data Center High Availability for details.

Table 14-1 shows Cisco products leveraged in the DCAP topology for achieving key IT objectives in the areas of Reliability, Availability, and Serviceability (RAS).

Table 14-1 Cisco Products That Leverage IT RAS Objectives in Cisco DCAP 4.0

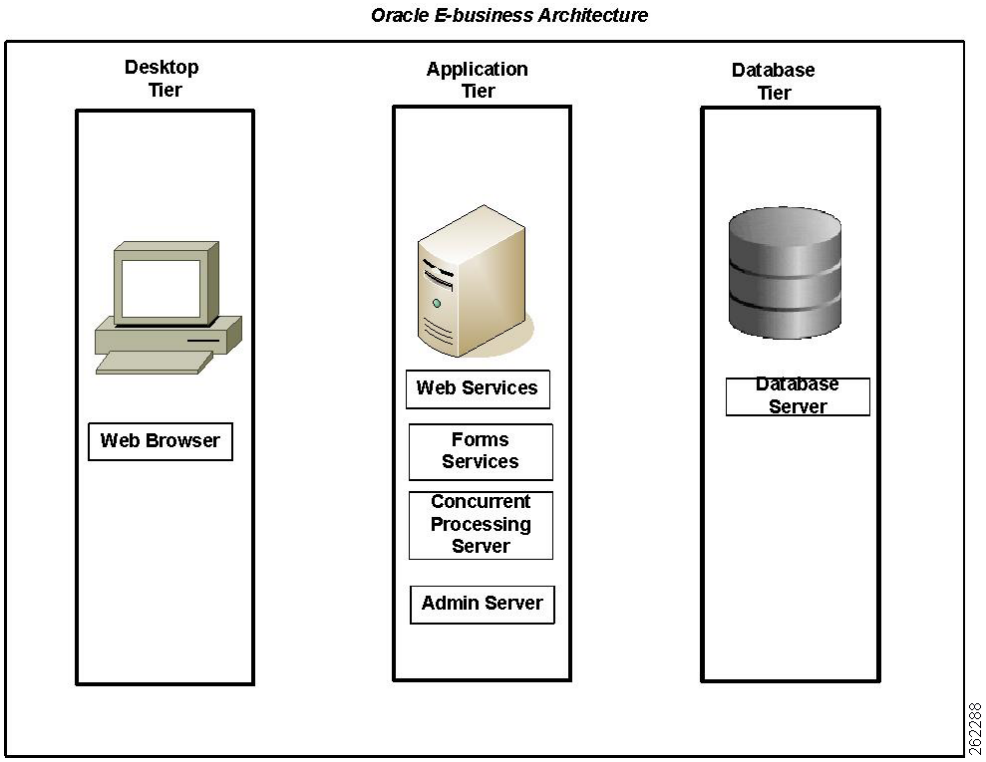
RAS Features	Cisco Products
Application Service Virtualization	ACE, GSS,CSM
Application Load Balancing	ACE, CSM,GSS
Application Optimization	WAAS
Business Resiliency	MDS,GSS
High Availability	MDS, Catalyst6500

E-Business Suite Architecture

Oracle Applications Architecture is a framework for multi-tiered, distributed computing that supports Oracle Applications products. In this model various Services/Servers are distributed among 3 tiers. The three-tier architecture that comprises an E-Business Suite installation is made up of Database Tier, which supports and manages Oracle Database, the Application Tier, which supports and manages various Application components and is sometimes known as middle tier, and the Desktop Tier, which provides the user interface via an add-on component to a standard web browser

The Oracle Application Architecture (Figure 14-1) separates logically the Desktop, Application and Database Tiers. In enterprise deployments, each tier can consist of one or more physical hosts to meet required High Availability, Scalability and Performance goals.

Figure 14-1 Oracle E-Business Suite Architecture



Desktop Tier

The Desktop Tier represents clients on the Internet or Intranet accessing the Application. An interface is provided through HTML for HTML-based applications, and via a Java applet in the Web browser for the traditional Forms-Based Applications.

Application Tier

The Application Tier has a dual role: hosting the various servers and service groups that process the business logic, and managing communication between the Desktop and Database Tiers. Figure 14-1

shows the four service groups that comprise the basic Application Tier for Oracle Applications

- Web Server
- Forms Server
- Concurrent Processing Server
- Admin Server

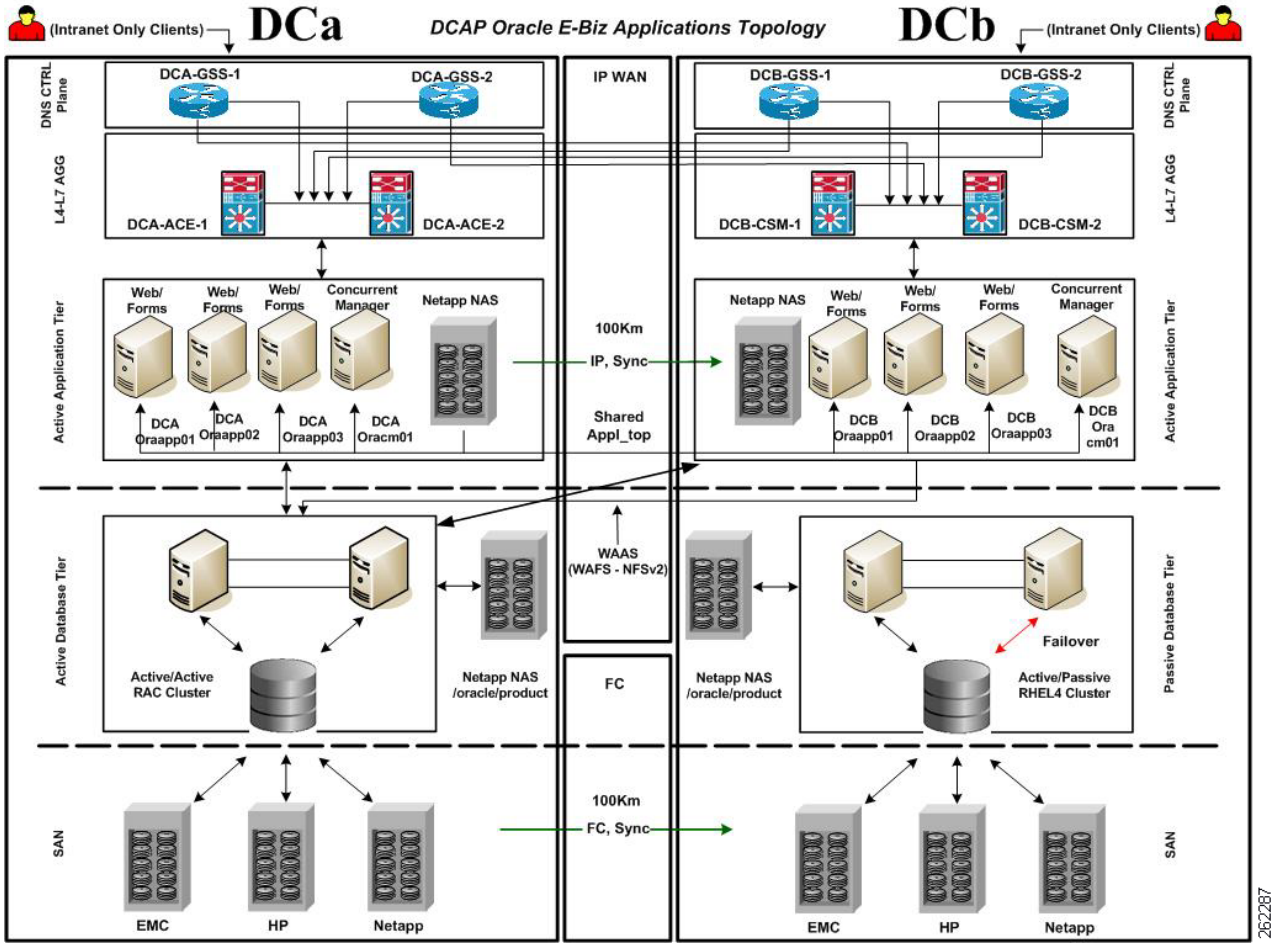
Database Tier

The Database Tier contains the Oracle Database Server, which stores all the data maintained by Oracle Applications. Servers in the Application Tier communicate with DB servers to process client requests.

DCAP Oracle E-Business Topology

Figure 14-2 shows the DCAP implementation of Oracle E-Business Suite in Active/Active Hybrid configuration across the two data centers DCa and DCb. E-Business Suite is installed in a multi-node configuration where the Application Tier is shared across the two data centers, making it active/active.The Database Tier is active in only one Data Center (DCa) with data replicating synchronously to the second Data Center (DCb) making it active/passive. This design also leverages integrated network services including Application load balancing, application optimization and SAN extension capabilities for synchronous data replication. This section details the DCAP implementation for each of logical Tiers of E-Business Suite.

Figure 14-2 Oracle E-Business Suite Topology



Desktop Tier

Intranet clients shown in the topology represent the Desktop Tier. Clients for the Oracle Application are located in three branch offices. Table 14-2 shows this branch configuration. Oracle clients use a Web browser with the HTTP protocol to access the applications URL at http://wwwin-oefin.gslb.dcap.com. All client traffic is optimized by WAAS. All clients are located on the DCAP intranet; no Internet clients have access, and therefore no advanced services like firewalls or proxies are needed.

Table 14-2 Cisco DCAP 4.0 Oracle Branch Configuration

Data Center	Branch1: T3(45Mbit/sec)	Branch2: T1(1.5Mbit/sec)	Branch3:T1 (1.5Mbit/sec)
DCa	Latency: 5msec	Latency: 16msec	Latency: 69msec
DCb	Latency: 6msec	Latency: 17msec	Latency: 70msec

Cisco WAAS is able to optimize all Oracle traffic sent on HTTP port 8000 with the given latencies in the table above. TCP Flow Optimization (TFO), LZ compression and Data Redundancy Elimination (DRE) all played a part in optimizing the Load Runner-generated traffic for various E-Business Suite transactions.

Aggregation Tier

All four GSS devices (two GSS devices at each data center) provided global server load balancing and disaster recovery for all of the oracle clients at each of the three branch locations. All four GSS devices were authoritative for the domain wwwin-oefin.gslb.dcap.com. All four GSS devices provided health checks for the Oracle application which was virtualized on ACE at DCa or CSM at DCb. The GSS's maintained the health of the Oracle applications running behind ACE in Dca or CSM in DCb. A client DNS request arrives at each of the four GSS devices by means of name server forwarding via the client's local branch name server. Once the DNS request arrives at one of the four GSS devices, that GSS's job is to hand out the Virtual IP Address of a VIP for the wwwin-oefin.gslb.dcap.com domain which lives at both DCa and DCb. At the time that the DNS query arrives at one of the four GSSdevices, all four are already are aware which VIPs at each data center (VIP on ACE at DCa or the VIP on CSM in DCb) are alive and available based on the GSS probe status. The GSS will then hand out the appropriate VIP-based first on the health and availability of the VIP at each data center which is a direct correlation to the Oracle applications health for which the ACE and CSM is virtualizing. Second, the GSS hands out the appropriate VIP based on the load balancing algorithm chosen by the administrator. The load balancing algorithm chosen by the administrator on the GSS is one that chooses the appropriate Virtual IP Address on the ACE in DCa or the CSM in DCb. The different types of load balancing for the GSS DNS rule on the GSS devices are as follows: Round Robin, Weighted Round Robin, Ordered List, Least Loaded and Hashed. Our tests used both Round Robin and Weighted Round Robin.

Refer to Volume 6: Global Site Selector (GSS) for details on how GSS is configured.

The ACE modules which are connected into each of the aggregation switches at DCa provide a level of virtualization and load balancing for the Oracle E-Business suite applications for which it is providing services. Each ACE has two specific policies configured in order to provide services for the Oracle E-Business Suite applications. The first policy is a policy type HTTP redirect.

```
policy-map multi-match ORACLE_TCP_TRAFFIC
 class REDIRECT_VIP_L4
    loadbalance vip inservice
    loadbalance policy REDIRECT_TCP:80_TO_TCP:8000
    loadbalance vip icmp-reply
    loadbalance vip advertise

class-map match-all REDIRECT_VIP_L4
  2 match virtual-address 101.1.33.50 tcp eq www

policy-map type loadbalance first-match REDIRECT_TCP:80_TO_TCP:8000
  class ONLY_VALID_HTTP_FOR_REDIRCT_CANDIDATE
    serverfarm REDIRECT_PORT_80_TO_PORT_8000

class-map type http loadbalance match-all ONLY_VALID_HTTP_FOR_REDIRCT_CANDIDATE
  2 match http header Host header-value "wwwin-oefin.gslb.dcap.com"
  3 match http url / method GET
```

The policy above redirects a client request that is destined for TCP port 80 (HTTP) to be issued an HTTP 302 redirect for TCP port 8000. This is the first policy that will be matched for incoming HTTP requests to the VIP on the ACE in DCa.

```
policy-map multi-match ORACLE_TCP_TRAFFIC
  class ORACLE_L4
    loadbalance vip inservice
```

```
    loadbalance policy GO_TO_WAE_FARM
    loadbalance vip icmp-reply

class-map match-any ORACLE_L4
  2 match virtual-address 101.1.33.50 tcp eq 8000

policy-map type loadbalance first-match GO_TO_WAE_FARM
  class class-default
    serverfarm WAE_FARM backup ORAAPP_ORACLE_FARM
```

The policy above receives HTTP requests resolving to the Fully-Qualified Domain Name (FQDN), wwwin-oefin.gslb.dcap.com, on TCP port 8000 for which the first policy redirected the client to. This policy will match a client request that arrives at the ACE on TCP port 8000 and load balance the request to one of the WAE's in the serverfarm WAE_FARM.

```
policy-map multi-match OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS
  class VIA_WAE_FARM_L4
    loadbalance vip inservice
    loadbalance policy ORAAPP_ORIGIN_SERVERS
    loadbalance vip icmp-reply

class-map match-any VIA_WAE_FARM_L4
  2 match virtual-address 101.1.33.50 tcp any

policy-map type loadbalance first-match ORAAPP_ORIGIN_SERVERS
  class class-default
    sticky-serverfarm sticky-ace-cookie
    insert-http SRC_IP header-value "%is"
```

The policy above receives TCP traffic on any TCP port sourced from the WAE serverfarm on VLAN1135. This policy will match any TCP traffic sourced from the WAEs on VLAN1135 and destined to the VIP 101.1.33.50 and ensure that the client will be then load balanced to the origin servers via a different policy on ACE.

The CSMs which are connected into each of the Aggregation Layer switches at DCb provide a level of virtualization and load balancing for the Oracle E-Business Suite applications for which it is providing services. Each CSM has two specific Vservers configured in order to provide services for the Oracle E-Business Suite applications. The first Vserver is a redirect.

```
vserver WWWIN-REDIRECT
  virtual 201.40.30.51 tcp www
  serverfarm 80-TO-8000
  persistent rebalance
  inservice
```

The Vserver above redirects a client request destined for TCP port 80 (HTTP) to another vserver listening on TCP port 8000. This is the first Vserver that will be matched for incoming HTTP requests.

```
vserver WWWIN-OEFIN
  virtual 201.40.30.51 tcp 8000
  vlan 301
  serverfarm ORACLE-ALL
  advertise active
  sticky 30 group 70
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  domain DCb-csm-1
  inservice
```

The above Vserver receives HTTP requests (wwwin-oefin.gslb.dcap.com) on TCP port 8000 for which the first Vserver redirected the client to. This Vserver will match a client request that arrives at the CSM on TCP port 8000 and load balance the request to one of the servers in the serverfarm ORACLE-ALL. This Vserver will also create a sticky entry in the sticky database that is based on the client's source IP address.

All four GSS devices communicate with the both the ACE in DCa as well as the CSM in DCb. All four GSS devices must be able to reach the ACE modules at DCa as well as the CSMs at DCb in order to understand their health and availability. This function is called a keepalive or probe. The following keepalive methods are used in the DCAP topology

- KAL-AP – Uses a UDP transport where the GSS interrogates an ACE/CSM in order to obtain load information on a particular VIP/Vserver or specific rule. KAL-AP keepalive type can be configured for either KAL-AP by "TAG" or KAL-AP by VIP.
- HTTP Head – GSS sends the VIP on the ACE/CSM HTTP Head request looking for a 200 OK server response code.
- TCP – GSS sets up a 3-way TCP handshake to the VIP on ACE/CSM in order to verify the TCP socket is open.

KAL-AP by TAG was used for the keepalive type between all four GSS devices and the CSMs at DCb.

It is extremely important to understand the different TCP and UDP ports that the GSS devices and the Global Site Selector Master (GSSM) use for keepalive functionality in order to help plan network topologies and positioning of the GSS devices and GSSMs.

In the topology, configuring health probes to the real servers allows you to determine if the real servers are operating correctly.

The health of a real server is categorized as follows:

- Active—The real server responds appropriately.
- Suspect—The real server is unreachable or returns an invalid response. The probes are retried.
- Failed—The real server fails to reply after a specified number of consecutive retries.

The GSS is notified and the ACE CSM adjusts incoming connections accordingly. Probes continue to a failed server until the server becomes active again. The probes used in the testing were HTTP type probes which logged into the server resource /oa_servlets/AppsLogin via HTTP with the appropriate username and password.

The health probes used on the ACE and the CSM are as follows:

On the ACE, an HTTP probe assigned to the application server named "ORACLE_WEB_PAGE_CHECK".

This probe creates a TCP 3-way handshake and then initiates an HTTP GET request for the Universal Reourcde Identifier (URI) /oa_servlets/AppsLogin sending the authentication credentials of Username: sysadmin and password: sysadmin. Upon receiving an HTTP 200 OK server response code from the application server, the ACE assumes the application is operational and makes the server available as a resource.

```
probe http ORACLE_WEB_PAGE_CHECK
  port 8000
  interval 2
  faildetect 1
  passdetect interval 2
  credentials sysadmin sysadmin
  request method get url /oa_servlets/AppsLogin
  expect status 200 200
```

```
dca-agg-1-ace-1/c2# show probe ORACLE_WEB_PAGE_CHECK

 probe       : ORACLE_WEB_PAGE_CHECK
 type        : HTTP, state : ACTIVE
----------------------------------------------
   port      : 8000    address     : 0.0.0.0          addr type  : -
   interval  : 2       pass intvl  : 2                pass count : 3
   fail count: 1       recv timeout: 10
                       -------------------- probe results --------------------
   probe association   probed-address  probes     failed     passed     health
   ------------------- ---------------+----------+----------+----------+-------
   serverfarm  : ORAAPP_ORACLE_FARM_WAAS_CONTENT
     real      : ORAAPP01[0]
                       101.1.33.16     57         0          57         SUCCESS
     real      : ORAAPP02[0]
                       101.1.33.5      57         0          57         SUCCESS
     real      : ORAAPP03[0]
                       101.1.33.47     57         0          57         SUCCESS
```

On the CSM an HTTP probe is assigned to the application server named “ORACLE”. This probe creates a TCP 3-way handshake and then initiates an HTTP GET request for the URI /oa_servlets/AppsLogin along with the HTTP Header “I_AM_CSM” and the authentication credentials of Username: sysadmin and password: sysadmin. Upon receiving an HTTP 302 Server Response Code from the application server, the CSM assumes the application is operational and makes the server available as a resource.

```
probe ORACLE http
  credentials sysadmin  sysadmin
  header I_AM_CSM
  request method get url /oa_servlets/AppsLogin
  expect status 302
  interval 2
  retries 1
  failed 2
  port 8000

DCb-ss-1#show mod csm 2 probe name oracle detail
probe           type    port  interval retries failed  open   receive
-------------------------------------------------------------------
ORACLE          http    8000  2        1       2       10     10
 Probe Credentials:
  Username: sysadmin             Passwd: sysadmin
 Probe Request:  GET        /oa_servlets/AppsLogin
 HTTP Headers:
  I_AM_CSM
 Expected Status Codes:
  302
 real                  vserver         serverfarm      policy          status
 ---------------------------------------------------------------------------
 201.1.33.47:8000      WWWIN-CLEAR     ORACLE-ALL      (default)       OPERABLE
 201.1.33.16:8000      WWWIN-CLEAR     ORACLE-ALL      (default)       OPERABLE
 201.1.33.5:8000       WWWIN-CLEAR     ORACLE-ALL      (default)       OPERABLE
```

Application Tier

The Application Tier in DCa of the four application hosts dcap-dca-oraapp01 (virtual host), dcap-dca-oraapp02 (HP blade server), dcap-dca-orapp03(virtual vost) and dcap-dca-oracm01 (HP blade servers). The hosts dcap-dca-oraapp01,dcap-dca-orapp02 and dcap-dca-oraapp03 are configured to provide front-end connectivity functions servicing Web and Forms services. Application hosts in data center A are configured behind the ACE to provide load balancing capabilities while at the same time providing high availability for services within data center A. The setup is similar in DCb where hosts

dcap-DCb-oraapp01 (virtual host), dcap-DCb-oraapp02 (HP blade server) and dcap-DCb-oraapp03(virtual host) provide Web and forms services and these application hosts are configured behind CSM to provide load balancing capabilities while at the same time providing high availability for services within data center B.

The hosts dcap-dca-oracm01 and dcap-DCb-oracm01 provide services to run Concurrent Manager/Batch jobs. These Concurrent Manager hosts are configured using the Parallel Concurrent Processing (PCP) feature thus providing capability of load balancing and high availability for batch processing jobs.

The ACE, in the Aggregation Layer of DCa, and the CSM, in the Aggregation Layer of DCb, provide load balancing between the three web hosts in each data center by means of a virtual IP (VIP). The VIPs for each data center are in different networks since there is no L2 adjacency between the data centers and no advanced capabilities like route health injection are being used. In DCAP deployment, Oracle 11i E-Business Suite is initially installed on a single application front end server using the standard Oracle E-Business Suite installation tool. Then, the Oracle Auto Configuration utility is used to configure Oracle Applications to leverage the VIP on either the ACE or the CSM. Table 14-3 highlights key changes required in Context file for each of the Application hosts to accommodate integration of ACE and CSM with E-Business Suite.

Table 14-3 Context File Changes for ACE and CSM Integration

Variable Name	Current Values	Changed Value
s_webentryhost	dcap-dca-oraapp01 DCa App Node1 dcap-dca-or aapp02-DCa App Node2 dcap-dca-oraapp03 DCa App Node3	wwwin-oefin (VIP on ACE)
	dcap-DCb-oraapp01 DCb App Node1 dcap-DCb-oraapp02 DCb App Node 2 dcap-DCb-oraapp03 DCb App Node 3	wwwin-oefin (VIP on CSM)
s_active_webport	8000	Bind port 8000 to port 80
S_webentry_domain	Dcap.com	Gslb.dcap.com (Domain name associated with VIP)
s_login_page	http://dcap-dca-oraapp01.dcap.com:8000/oa_servlets/AppsLogin - DCa App Node 1 http://dcap-dca-oraapp02.dcap.com:8000/oa_servlets/AppsLogin - DCa App Node 2 http://dcap-dca-oraapp01.dcap.com:8000/oa_servlets/AppsLogin - DCa App Node 3 http://dcap-DCb-oraapp01.dcap.com:8000/oa_servlets/AppsLogin - DCb App Node 1 http://dcap-DCb-oraapp01.dcap.com:8000/oa_servlets/AppsLogin - DCb App Node 2 http://dcap-DCb-oraapp01.dcap.com:8000/oa_servlets/AppsLogin - DCb App Node 3	

Shared APPL_TOP

A traditional multi-node installation of 11i E-Business Suite requires each Application host to maintain its own Application Tier file system consisting of APPL_TOP and COMMON_TOP directories and the Application Tier technology stack file system (iAS and 8.0.6 Oracle homes). In the DCAP topology the "Shared Application File System" architecture is implemented with the ability to share the APPL_TOP file system and Application Tier tech stack filesystem. All the Application Tier files are installed on single NetApp filer cluster volume (/apps/oefin) located in data center A and mounted using NFSv2 over TCP across all the Application hosts in DCa and DCb. To enable failover to data center B, the cluster volume is synchronously replicated over an IP WAN link using synchronous SnapMirror to a NetApp filer cluster in data center B.

Utilizing a shared APPL_TOP and shared Application Tier file system is key component for Active/Active Application tier across data centers. Other benefits include

- Flexibility to add additional nodes to existing installation, thereby providing greater resiliency to node failure or to support additional user capacity.
- Software patches only need to be applied to one Application Tier node for the effects to be visible on all other nodes that share the file system. This will minimize the duration of planned maintenance downtime.

Note Refer to Note id: 233428.1 http://metalink.oracle.com for details on how to enable shared appl_top.

Forms Deployment mode

Oracle Forms can be deployed in Servlet Mode or Socket Mode. In Servlet Mode there is a java servlet called the Forms Listener Servlet that manage the communication between the Forms Java Client and OracleAS Forms Services. This architecture operates through the HTTP server port alone and does not need extra ports to handle communication between the client and the application server.

In the current DCAP implementation, "Servlet Mode" is enabled. On the ACE in DCa, sticky was also enabled in order to ensure that the client remains stuck to the same server for the duration of the session. In DCa, on the ACE, sticky is based on an ACE inserted cookie in order to provide persistence.

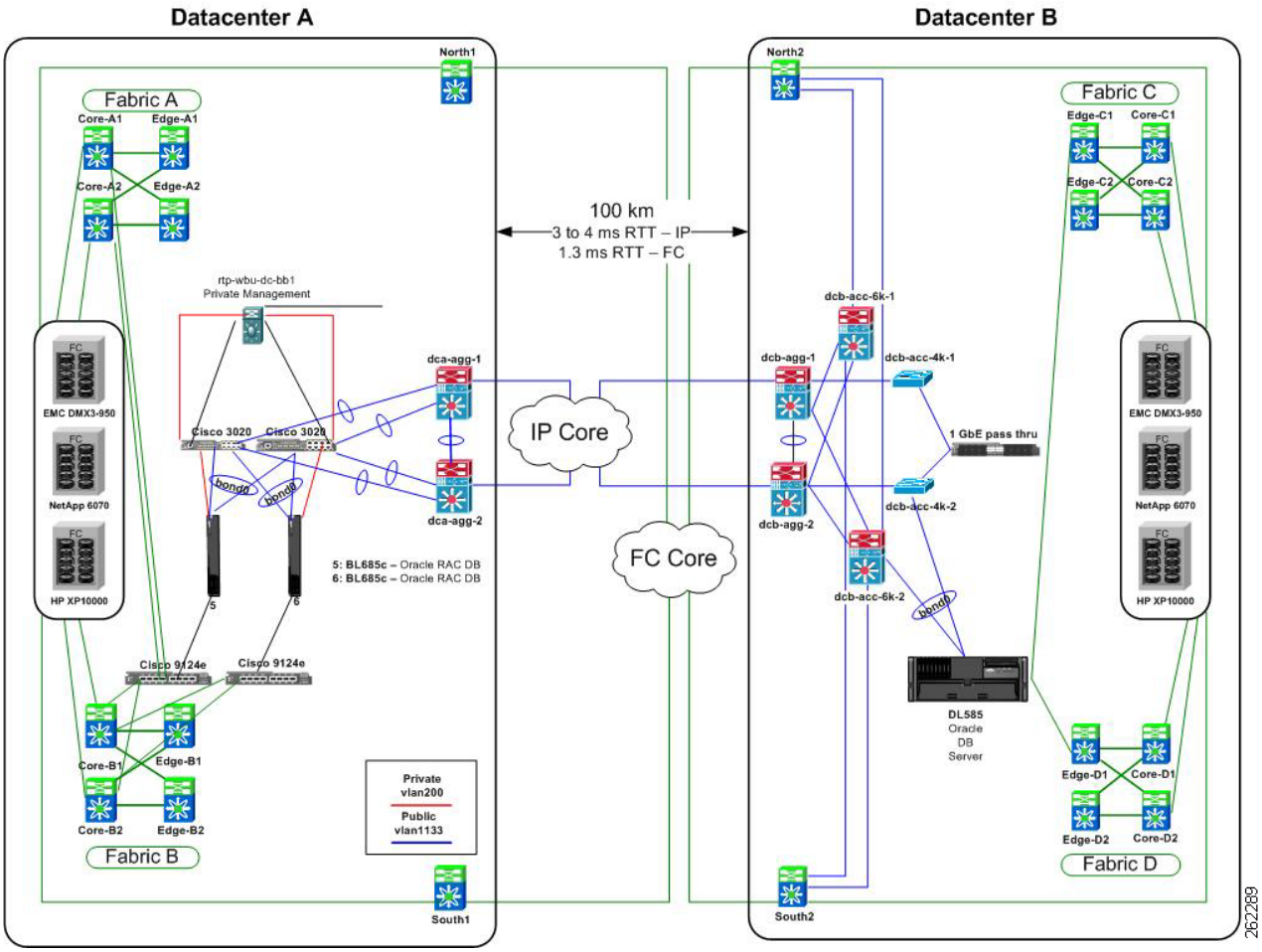
In the DCb, sticky must be enabled on the CSM to ensure that the user that was load balanced to the application server for Oracle Forms. In the testing, the method of sticky used was "sticky source IP Address ."

Database Tier

The Database Tier in DCa consists of active/active 10gR2 Real Application Cluster (RAC). The cluster is configured using two RedHat Enterprise Linux update 4 hosts (HP BL685 blade servers). The shared storage for RAC is configured using Oracle Automatic Storage Management (ASM). The database, called OEFIN, is normally located in DCa on SAN storage from EMC, HP, and NetApp. This storage uses each vendor's synchronous replication method (SRDF/S for EMC, Continuous Access XP Synchronous for HP, and synchronous SnapMirror for NetApp) over an extended fibre channel link to ensure an identical copy of the database is available for failover in data center B.

RAC configuration provides horizontal scalability for increased capacity and high availability for Database Tier.

Figure 14-3 Cisco DCAP 4.0 Real Application Clusters (RAC) Topology



The Database Tier in DCb consists of two RedHat Enterprise Linuxupdate 4 hosts (DL585 servers) configured in Active/Passive mode using RedHat Enterprise Linux update 4 Active/Passive Cluster. RAC deployment for this failover Data Center is deferred to a future phase.

Note Refer to Note ID: 362135.1 http://metalink.oracle.com for information on how to configure Oracle Applications Release 11i with 10gR2 RAC and ASM.

DCAP Oracle E-Business Environment

The following section provides an overview of the hardware and software used in the DCAP E-Business Suite environment.

Hardware: Application Tier VMware Deployment

VMWare ESX server version 3.0.2 was deployed on two clustered HP c-Class blade servers in each data center. The Data Center A deployment used BL480c dual-processor, quad-core, 2.6Ghz Intel powered servers with 16GB of RAM. The ESX OS was installed on a SAN Boot LUN and the VM storage was provisioned on a SAN datastore. To provide network high availability at the ESX level each server was configured to use four bonded Gigabit Ethernet NIC's for inband connectivity and two bonded Gigabit Ethernet NIC's for management. Oracle 11i was deployed by provisioning a single virtual host on each ESX server. Each Oracle VM was provisioned 4GB RAM and 2CPUs. By following the ESX design implementation guides, the VMware servers' network settings were configured to allow VMotion, providing yet level of resiliency. The DCb deployment was configured in a similar manner to DCa with the main difference being the use of BL465c Dual-Core, 2.6Ghz, AMD servers with 8GB of total RAM.

Application Tier Server Deployment

- 2 HP BL465c with dual core AMD Opteron 1.8GHz CPU and 4GB of RAM
- 2 HP BL460c with dual core Intel 1.6GHz CPU and 4GB of RAM

Database Tier

- 2 HP BL685's with dual AMD opteron 2.8GHz CPU and 32GB of RAM in DCa
- 2 HP DL585's with dual AMD opteron 2.8GHz CPU and 32GB of RAM in DCb

Software

Oracle E-Business Suite

- Oracle11i – 11.5.10.2

Techstack Patch Level

- ATG RUP4

Oracle Database

- Oracle 10gR2 RAC

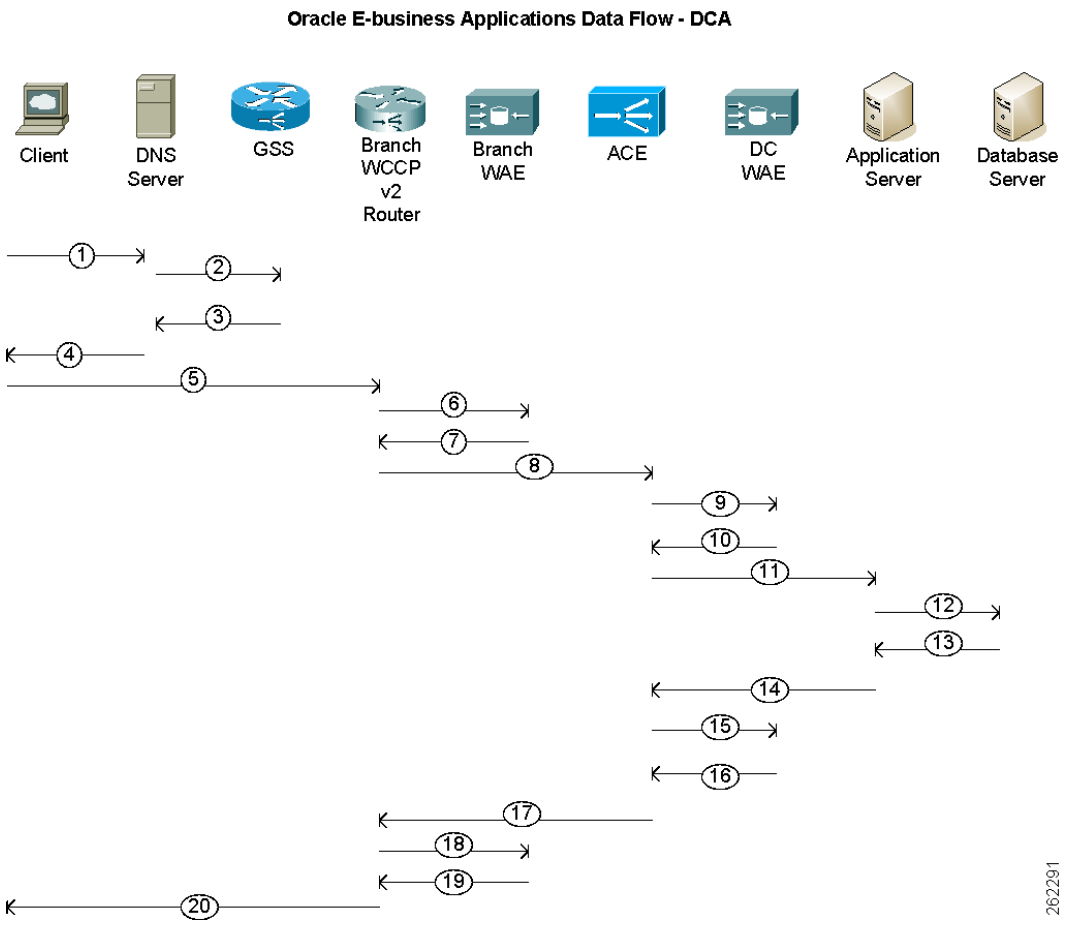
Operating System

- RHEL4 Update 4 32bit for Application Tier
- RHEL4 Update4 64bit for Database Tier

Application Traffic Flow in Data Center A

Figure 14-4 provides an overview of the application data flow from a branch client to the E-Business Suite application showing the major components. It details the data flow from the client, located at the branch office, connecting to the E-Business Suite application residing in the data center.

Figure 14-4 Cisco DCAP 4.0 Application Traffic Flow DCa



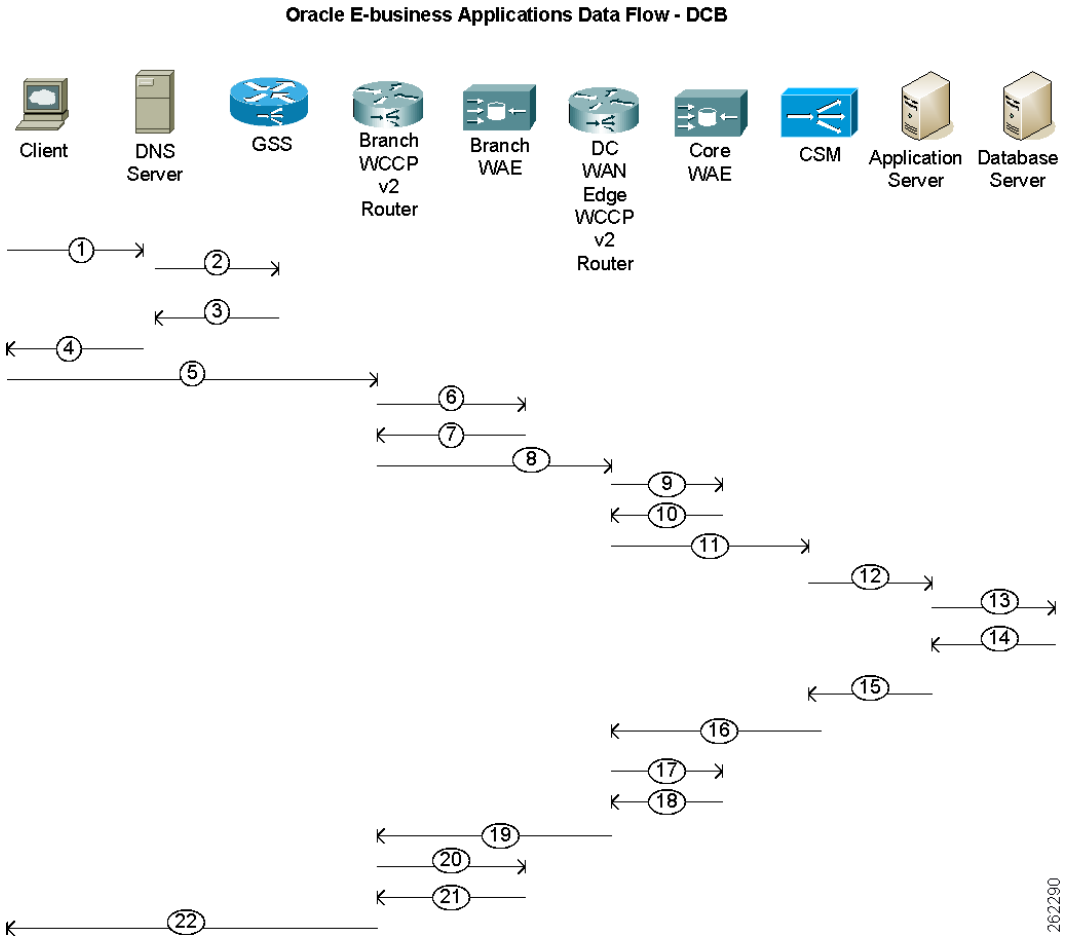
1. Client makes DNS requests for to Branch Name Server
2. Branch Name Server NS Forwards DNS query for to GSS
3. GSS responds with Authoritative Answer of 101.1.33.50 (VIP at DCa)
4. Branch Name Server responds with VIP from DCa (TTL of 5sec) to Client
5. Client sends HTTP request to ACE
6. Branch router intercepts client request via WCCP interception and forwards to branch WAE
7. Branch WAE responds back to branch router with TCP option 21 (0x21) set
8. Branch router forwards request to VIP on ACE (101.1.33.50)
9. ACE forwards request transparently to WAE
10. WAE responds back to ACE

11. ACE forwards request to Application server
12. Application server forwards data requests to Database server
13. Database server responds with required data back to Application server
14. Application server forwards response back to ACE
15. ACE response back to WAE
16. WAE responds back to ACE transparently
17. ACE forwards request back to branch router
18. Branch router forwards request back to branch WAE
19. Branch WAE forwards request back to branch router
20. Branch router forwards HTTP response back to client.

Application Traffic Flow in Data Center B

Figure 14-5provides an overview of the Application data flow from a branch client to the E-Business Suite application showing the major components. It details the data flow from the Client, located at the branch office, connecting to the E-Business Suite application residing in the data center.

Figure 14-5 Cisco DCAP 4.0 Application Traffic Flow DCb



1. Client makes DNS requests for to Branch Name Server
2. Branch Name Server NS Forwards DNS query for to GSS
3. GSS responds with Authoritative Answer of 201.40.30.51 (VIP at DCb)
4. Branch Name Server responds with VIP from DCb (TTL of 5sec) to Client
5. Client sends HTTP request to CSM
6. Branch router intercepts client request via WCCP interception and forwards to branch WAE
7. Branch WAE responds back to branch router with TCP option 21 (0x21) set
8. Branch router routes the request to VIP at CSM
9. The WAN edge router intercepts the request via WCCP and forwards to core WAE
10. Core WAE responds back to DC WAN edge router

11. WAN edge router forwards the request to the destination VIP on CSM
12. CSM load balances requests to the Application server
13. Application server forwards data requests to Database server
14. Database server responds with required data back to Application server
15. Application server forwards response back to CSM
16. CSM forwards response back to WAN edge router.
17. The DC WAN edge router intercepts the request via WCCP and forwards to core WAE
18. Core WAE responds back to DC WAN edge router
19. The DC WAN edge router routes the request back to branch router
20. Branch router intercepts client request via WCCP interception and forwards to branch WAE
21. Branch WAE responds request back to branch router
22. Branch router forwards HTTP request back to client.

Oracle Failover/Failback Summary

Table 14-4 summarizes the results of failover and failback testing for Oracle E-Business Suite. These tests are conducted as part of data Center failover testing for each of storage vendor. Complete details for each of these tests are covered in Volume 11—Data Center High Availability.

Note Refer to Volume 11: Data Center High Availability for HA details.

Table 14-4 Oracle Failover/Failback Test Results

	Failover		Failback	
Vendor	**RPO**	**RTO**	**RPO**	**RTO**
EMC	0	14 min	0	21 min
HP	0	12 min	0	19 min
NetApp	0	14 min	0	19 min

Testing Summary

The Cisco DCAP 4.0 Oracle E-Buisness Suite application tests fall into two major categories. The Baseline Functionality tests, which are documented in this volume, were executed to verify the configuration and functionality of Oracle E-Business Suite integration with GSS, ACE, CSM, Active/Active hybrid mode and WAAS optimizations. Application Traffic was generated by clients from three branch offices to both data centers with WAAS and without WAAS.

The Cisco DCAP 4.0 High Availability, Volume 11 include failing a component in the network and characterizing the impact on Oracle application environment by the failure and failover to a redundant component and failback to the primary component once it is again operational. Key metrics in characterizing the application such as user experience and application throughput were used to quantify failover and failback function.

Note Refer to Volume 11: Data Center High Availability for HA details.

HP's Mercury Load Runner tool was leveraged to simulate application traffic. The Load Runner environment has one Controller located in DCB and three generators located at the three branch offices. Five business processes, Create_Project_forms, DCAP_Receivables, CRM_Manage_Role, DCAP_Create_User, and iProcurement (comprised of Oracle Forms and HTTP functionality), are identified to simulate real-time traffic. Details on functionality of each of these business processes are explained in the Appendix section. The following scenarios were tested to determine the average transaction response times per data center from each of the branch clients.

- 10 Simultaneous Users test (for each of 3 branch generators for DCa and DCb separately with WAAS turned ON and OFF)
- 150 Simultaneous Users test (global test with both DCa and DCb in the configuration with WAAS turned ON and OFF)
- 150 Simultaneous Users test for failover testing

Branch Comparison Summary Results

This section summarizes results from the testing conducted between individual branches offices to each of the Data centers. Graphs provide comparison on average transaction response times on non-optimized WAN without WAAS and improvements provided by WAAS solution. Please see the detailed test results for more information. Detailed optimization stats from WAAS can also be found in test results.

Figure 14-6 provides a comparison of Average Transaction Response times from Branch1 to DCa with WAAS and without WAAS.

Figure 14-7 provides a comparison of Average Transaction Response times from Branch2 to DCa with WAAS and without WAAS.

Figure 14-8 provides a comparison of Average Transaction Response times from Branch3 to DCa with WAAS and without WAAS.

Figure 14-9 provides a comparison of Average Transaction Response times from Branch1 to DCb with WAAS and without WAAS.

Figure 14-10 provides a comparison of Average Transaction Response times from Branch2 to DCb with WAAS and without WAAS.

Figure 14-11 provides a comparison of Average Transaction Response times from Branch3 to DCb with WAAS and without WAAS.

Figure 14-6 Cisco DCAP 4.0 Branch 1 DCa WAAS Comparison

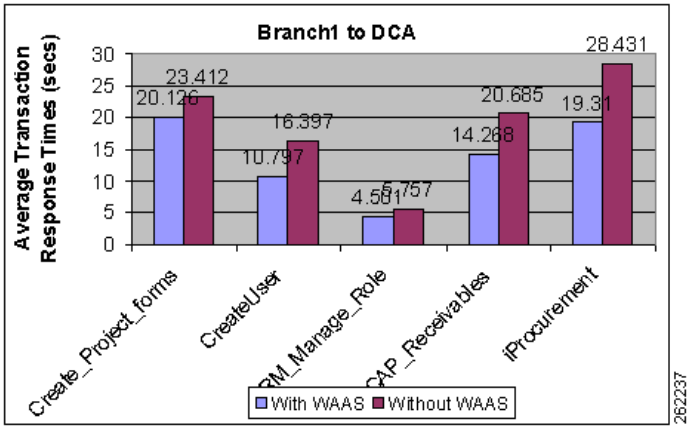


Figure 14-7 Cisco DCAP 4.0 Branch 2 DCa WAAS Comparison

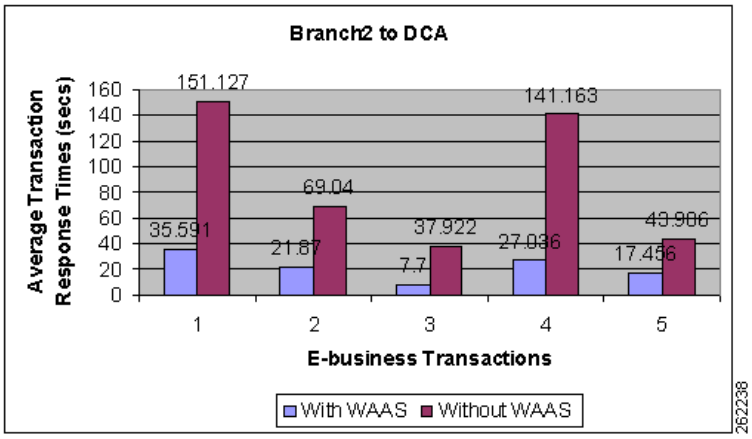


Figure 14-8 Cisco DCAP 4.0 Branch 3 DCa WAAS Comparison

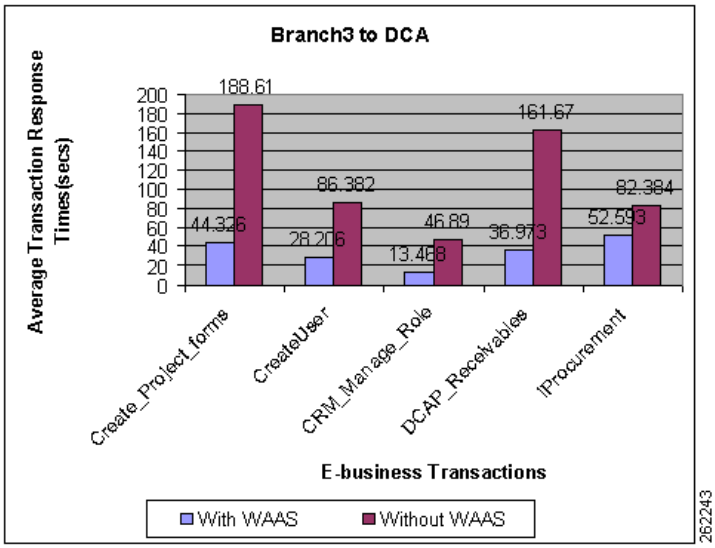


Figure 14-9 Cisco DCAP 4.0 Branch 1 DCb WAAS Comparison

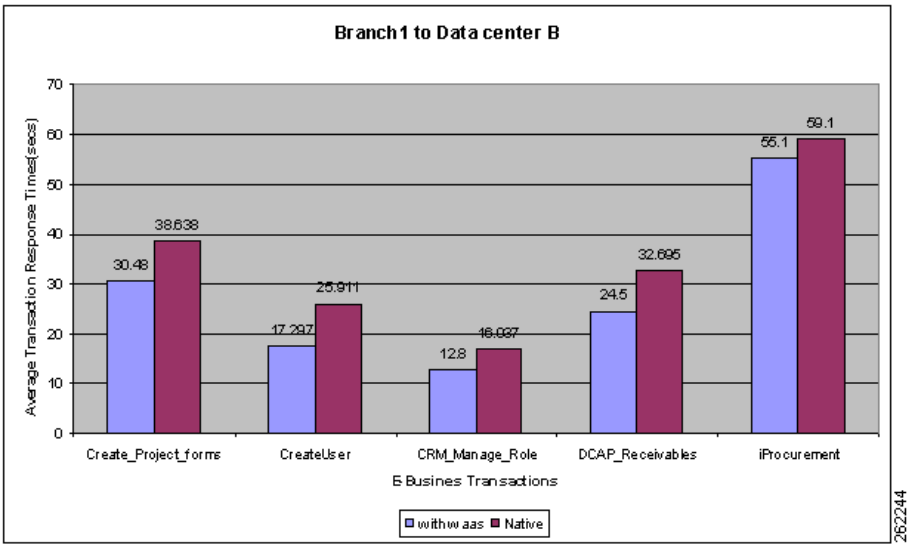


Figure 14-10 Cisco DCAP 4.0 Branch 2 DCb WAAS Comparison

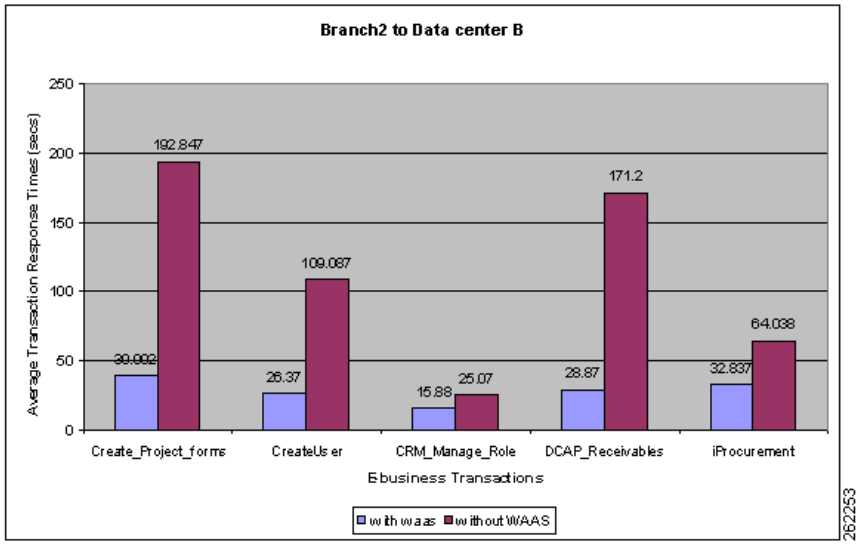
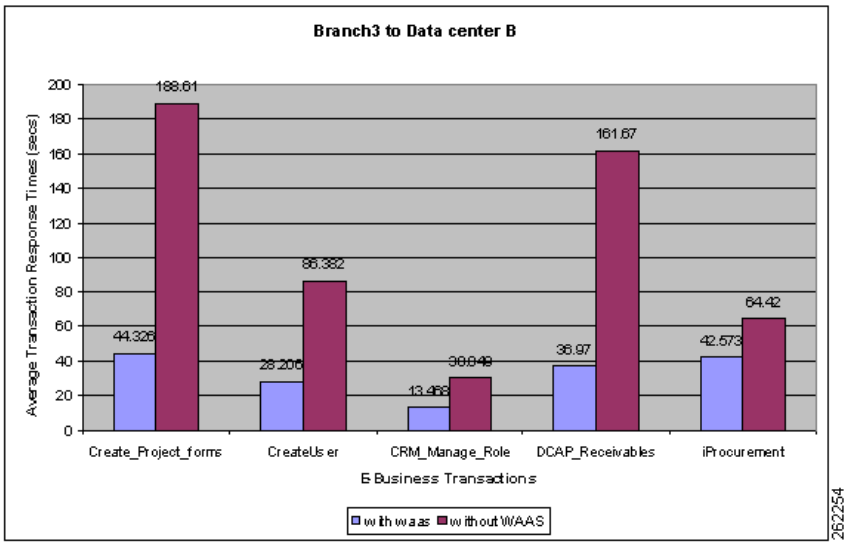


Figure 14-11 Cisco DCAP 4.0 Branch 3 DCb WAAS Comparison



Oracle Applications Configuration Details

Refer to Volume 12: Appendices and Volume 25: Oracle E-Business Configurations for Oracle configuration details. Oracle Vision Demo environment is installed in multi-node mode using Oracle Installation tool “RAPID INSTALL”. 11.5.10.2 by default is installed with database version 9iR2. The DB is upgraded to Oracle 10gR2 RAC and Oracle Applications are upgraded to latest Technology stack ATG.RUP4 patch set. The Application Tier and Database Tier have been enabled with Autoconfig.

Test Results Summary

Table 14-5 on page 14-21 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 14-5 on page 14-21 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 14-5 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Basic Functionality, page 14-22	n/a	1. Global Distribution of Oracle Application Traffic with WAAS 2. Global Distribution of Oracle Application Traffic without WAAS 3. Oracle Applications Traffic from Branch 1 to DCa with WAAS 4. Oracle Applications Traffic from Branch 1 to DCa without WAAS 5. Oracle Applications Traffic from Branch 1 to DCb with WAAS 6. Oracle Applications Traffic from Branch 1 to DCb without WAAS 7. Oracle Applications Traffic from Branch 2 to DCa with WAAS 8. Oracle Applications Traffic from Branch 2 to DCa without WAAS 9. Oracle Applications Traffic from Branch 2 to DCb with WAAS 10. Oracle Applications Traffic from Branch 2 to DCb without WAAS 11. Oracle Applications Traffic from Branch 3 to DCa with WAAS 12. Oracle Applications Traffic from Branch 3 to DCa without WAAS 13. Oracle Applications Traffic from Branch 3 to DCb with WAAS 14. Oracle Applications Traffic from Branch 3 to DCb without WAAS 15. Oracle E-Business Applications Environment Validation	

Test Cases

Functionality critical to global enterprises tested for this Cisco DCAP release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Basic Functionality

Baseline functionality is to verify configuration and functionality of Oracle E-business suite integration with GSS,ACE,CSM in Active/Active hybrid mode deployed across 2 Data Center. Tests also cover Application optimization using WAAS technology

This section contains the following topics:

Global Distribution of Oracle Application Traffic with WAAS

This test verified the functionality of the Oracle E-business Applications deployment across both Data Centers. This involved sending load runner based traffic from all the branch servers to both DCA and DCB

It was verified that the GSS distribution of client DNS queries worked as expected across both Data Centers. Once the GSS has distributed the queries it was verified that the ACE and CSM load balanced the connections across the application hosts as expected.

150 simultaneous users are simulated from all branch servers 1, 2 and 3 to both DCA and DCB. Simulated latency varied from 4ms to 70ms depending on where the traffic is originated from branches. The response times were then measured and performance results were quantified. During this test WAAS was accelerating the Oracle traffic.

Test Procedure

The procedure used to perform the Global Distribution of Oracle Application Traffic with WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from all the 3 Branches. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branches to the Application and Database hosts in both data centers.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool.From ACE, issue the following command to see the connection entries from the LoadRunner generated traffic to the VIP

```
show conn | inc 10.0..
```

From the CSM, issue the command, "show mod csm 2 conns vserver wwwin-oefin" and "show mod csm 2 sticky" in order to verify http traffic.

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS, CSM, ACE, and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands. On the CSM issue the **clear module csm *module*** connections, **clear module csm *module*** counters, **clear module csm *module*** sticky all commands. On the WAE devices issue the **clear statistics all** command.

From the ACE, issue the following commands in order to clear the appropriate counters:

```
clear service-policy ORACLE_TCP_TRAFFIC
clear service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS
clear sticky database all
clear serverfarm REDIRECT_PORT_80_TO_PORT_8000
clear serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
clear serverfarm WAE_FARM
clear stats all
```

Step 5 Initiate the Load Runner generated traffic which will run for approximately 1 hour.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the CSM that you see the expected behavior for connections, sticky entries and hits on the real servers by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
```

```
 sticky
show module csm

module
 reals sfarm

sfarm
 detailFrom these commands you should see an even distribution of connections to each real
server, each with it's own sticky entry based on source IP address. On ACE verify the
following commands: show service-policy ORACLE_TCP_TRAFFIC detail show service-policy
OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail show serverfarm WAE_FARM show serverfarm
ORAAPP_ORACLE_FARM show stats http show stats loadbalance show stats sticky show
service-policy ORACLE_TCP_TRAFFIC | inc drop show service-policy
OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop show stats inspect
```

Step 8 Verify the traffic is being accelerated by the WAE devices at the branches by issuing the **show tfo connection summary** command.

From the output of the command you should see all client to application connections established and being fully optimized.

Step 9 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will display the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 10 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 11 Verify on the CSM that you observe the proper expected results by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailCompare this with the pre test output and verify you see the expected changes from
the pre test results.
```

Verify on the ACE the following: show service-policy ORACLE_TCP_TRAFFIC detail show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail show serverfarm WAE_FARM show serverfarm ORAAPP_ORACLE_FARM show stats http show stats loadbalance show stats sticky show service-policy ORACLE_TCP_TRAFFIC | inc drop show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop show stats inspect

Step 12 Verify the statistics on the WAE device at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
```

```
show statistics dre
```

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that WAAS will accelerate transaction response times.
- We expect WAAS to optimize traffic from the Application hosts in DCB accessing the Database in DCA.
- We expect that Oracle transactions submitted through Load Runner from all branches across both Data Centers will be completed successfully.
- We expect that GSS will direct the Application traffic to both Data Centers as per design criteria.
- We expect that ACE and CSM will load balance the connections across the application hosts in their respective Data Centers.

Results

Global Distribution of Oracle Application Traffic with WAAS passed.

Global Distribution of Oracle Application Traffic without WAAS

This test verified the functionality of the Oracle E-business Applications deployment across both Data Centers. This involved sending load runner based traffic from all the branch servers to both DCA and DCB

It was verified that the GSS distribution of client DNS queries worked as expected across both Data Centers. Once the GSS has distributed the queries it was verified that the CSM load balanced the connections across the application hosts as expected.

150 simultaneous users are simulated from all branch servers 1, 2 and 3 to both DCA and DCB. Simulated latency varied from 4ms to 70ms depending on where the traffic is originated from branches. The response times were then measured and performance results were quantified. During this test WAAS was is disabled.

Test Procedure

The procedure used to perform the Global Distribution of Oracle Application Traffic without WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from all the 3 Branches. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branches to the Application and Database hosts in both data

centers.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From ACE, issue the following command to see the connection entries from the LoadRunner generated traffic to the VIP

```
show conn | inc 10.0..
```

From the CSM, issue the command, "show mod csm 2 conns vserver wwwin-oefin" and "show mod csm 2 sticky" in order to verify http traffic.

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS, CSM and ACE devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands. On the CSM issue the **clear module csm *module*** connections, **clear module csm *module*** counters, **clear module csm *module*** sticky all commands.From the ACE, issue the following commands in order to clear the appropriate counters:

```
clear service-policy ORACLE_TCP_TRAFFIC
clear service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS
clear sticky database all
clear serverfarm REDIRECT_PORT_80_TO_PORT_8000
clear serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
clear serverfarm WAE_FARM
clear stats all
```

Step 5 Initiate the Load Runner generated traffic which will run for approximately 1 hour.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the CSM that you see the expected behavior for connections, sticky entries and hits on the real servers by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailFrom these commands you should see an even distribution of connections to each real
server, each with it’s own sticky entry based on source IP address. On ACE verify the
following commands: show service-policy ORACLE_TCP_TRAFFIC detail show service-policy
OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail show serverfarm WAE_FARM show serverfarm
ORAAPP_ORACLE_FARM show stats http show stats loadbalance show stats sticky show
service-policy ORACLE_TCP_TRAFFIC | inc drop show service-policy
OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop show stats inspect
```

Step 8 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 9 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 10 Verify on the CSM that you observe the proper expected results by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailCompare this with the pre test output and verify you see the expected changes from
the pre test results.
```

Verify on the ACE the following: show service-policy ORACLE_TCP_TRAFFIC detail show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail show serverfarm WAE_FARM show serverfarm ORAAPP_ORACLE_FARM show stats http show stats loadbalance show stats sticky show service-policy ORACLE_TCP_TRAFFIC | inc drop show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop show stats inspect

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Transaction response times from the Application hosts in DCB accessing the Database in DCA to be longer as WAAS is disabled.
- We expect that majority of Oracle transactions submitted through Load Runner from all branches across both Data Centers will be completed successfully.
- We expect that, without the acceleration capabilities provided by WAAS, there will be timeout of incoming requests from remote branch offices.
- We expect that GSS will direct the Application traffic to both Data Centers as per design criteria.
- We expect that ACE and CSM will load balance the connections across the application hosts in their respective Data Centers.

Results

Global Distribution of Oracle Application Traffic without WAAS passed.

Oracle Applications Traffic from Branch 1 to DCa with WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch 1 to DCa.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the ACE load balanced the connections across the application hosts as expected.

The connection from branch to data center was simulated to have T3 bandwidth and 5 ms of latency. The response times were then measured and performance results were quantified. During this test WAAS was accelerating the Oracle traffic.

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 1 to DCa with WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From the ACE, issue the following command in order to verify the ACE's FT status: **show ft group summary** From the ACE, issue the following commands in order to clear the appropriate counters:

```
clear service-policy ORACLE_TCP_TRAFFIC
clear service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS
clear sticky database all
clear serverfarm REDIRECT_PORT_80_TO_PORT_8000
clear serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
clear serverfarm WAE_FARM
clear stats all
```

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the ACE that you see the connection entries from the LoadRunner generated traffic t o the VIP on ACEby issuing the following command.

```
show conn | inc 10.0..
```

Step 8 Verify the traffic is being accelerated by the WAE devices at the branch by issuing the **show tfo connection summary** command.

From the output of the command you should see all client to application connections established and being fully optimized.

Step 9 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 10 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 11 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 12 Verify on the ACE that you observe the proper expected results by issuing the following commands:

```
show service-policy ORACLE_TCP_TRAFFIC detail
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail
show serverfarm WAE_FARM
show serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP01 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP02 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP03 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show stats http
show stats loadbalance
show stats sticky
show service-policy ORACLE_TCP_TRAFFIC | inc drop
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop
show stats inspect Stop the syslog server and copy the log file to the results file.
```

Step 13 Verify the statistics on the WAE device at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect WAAS to accelerate transaction response times.
- We expect Oracle transactions submitted through Load Runner to be completed successfully.
- We expect the GSS to direct the Application traffic from Branch to Data Center.
- We expect the ACE to load balance the connections across the application hosts.
- We expect database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 1 to DCa with WAAS passed.

Oracle Applications Traffic from Branch 1 to DCa without WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch1 to DCA.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the ACE load balanced the connections across the application hosts as expected. It was also verified that the Database connections are distributed across both RAC instances

The connection from branch to data center was simulated to have T3(45mb/sec) bandwidth and 4ms of latency. The response times were then measured and performance results were quantified.During this test WAAS is disabled

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 1 to DCa without WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool.From the ACE, issue the following command in order to verify the ACE's FT status: **show ft group summary** From the ACE, issue the following commands in order to clear the appropriate counters:

```
clear service-policy ORACLE_TCP_TRAFFIC
clear service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS
clear sticky database all
clear serverfarm REDIRECT_PORT_80_TO_PORT_8000
clear serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
clear serverfarm WAE_FARM
clear stats all
```

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Follow these steps:

```
Verify on the ACE that you see the connection entries from the LoadRunner generated
traffic t o the VIP on ACE by issuing the following command.
```

Follow these steps:

```
show conn | inc 10.0..
```

Step 8 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 9 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 10 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 11 Verify on the ACE that you observe the proper expected results by issuing the following commands:

```
show service-policy ORACLE_TCP_TRAFFIC detail
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail
show serverfarm WAE_FARM
show serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP01 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP02 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP03 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show stats http
show stats loadbalance
show stats sticky
show service-policy ORACLE_TCP_TRAFFIC | inc drop
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop
show stats inspect Stop the syslog server and copy the log file to the results file.
```

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Oracle transactions submitted through Load Runner will be completed successfully.
- We expect transaction response times to be slightly higher since WAAS is disabled.
- We expect the GSS to direct the Application traffic from Branch to Data Center.
- We expect the ACE to load balance the connections across the application hosts.
- We expect the Database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 1 to DCa without WAAS passed.

Oracle Applications Traffic from Branch 1 to DCb with WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch1 to DCB.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the CSM load balanced the connections across the application hosts as expected. It was also verified that the Database connections are distributed across both RAC instances

The connection from branch to data center was simulated to have T3(45mb/sec) bandwidth and 6ms of latency. The response times were then measured and performance results were quantified. During this test WAAS was accelerating the Oracle traffic.

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 1 to DCb with WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From the CSM, issue the command, "show mod csm 2 conns vserver wwwin-oefin" and "show mod csm 2 sticky" in order to verify http traffic.

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS, CSM, and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands. On the CSM issue the **clear module csm *module*** connections, **clear module csm *module*** counters, **clear module csm *module*** sticky all commands. On the WAE devices issue the **clear statistics all** command.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the CSM that you see the expected behavior for connections, sticky entries and hits on the real servers by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
```

```
show module csm

module
 reals sfarm

sfarm
 detailFrom these commands you should see an even distribution of connections to each real
server, each with it's own sticky entry based on source IP address.
```

Step 8 Verify the traffic is being accelerated by the WAE devices at the branch by issuing the **show tfo connection summary** command.

From the output of the command you should see all client to application connections established and being fully optimized.

Step 9 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 10 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 11 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 12 Verify on the CSM that you observe the proper expected results by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailCompare this with the pre test output and verify you see the expected changes from
the pre test results.
```

Step 13 Verify the statistics on the WAE device at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect WAAS to accelerate transaction response times.
- We expect Oracle transactions submitted through Load Runner to be completed successfully.
- We expect the GSS to direct the Application traffic from Branch to Data Center.
- We expect the CSM to load balance the connections across the application hosts.
- We expect database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 1 to DCb with WAAS passed.

Oracle Applications Traffic from Branch 1 to DCb without WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch1 to DCB.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the CSM load balanced the connections across the application hosts as expected.It was also verified that the Database connections are distributed across both RAC instances

The connection from branch to data center was simulated to have T3(45mb/sec) bandwidth and 6ms of latency. The response times were then measured and performance results were quantified.During this test WAAS is disabled

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 1 to DCb without WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From the CSM, issue the command, "show mod csm 2 conns vserver wwwin-oefin" and "show mod csm 2 sticky" in order to verify http traffic.

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS, CSM, and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands. On the CSM issue the **clear module csm *module*** connections, **clear module csm *module*** counters, **clear module csm *module*** sticky all commands.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the CSM that you see the expected behavior for connections, sticky entries and hits on the real servers by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailFrom these commands you should see an even distribution of connections to each real
server, each with it's own sticky entry based on source IP address.
```

Step 8 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 9 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 10 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 11 Verify on the CSM that you observe the proper expected results by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailCompare this with the pre test output and verify you see the expected changes from
the pre test results.
```

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Oracle transactions submitted through Load Runner to be completed successfully.
- We expect the GSS to direct the Application traffic from Branch to Data Center.
- We expect the CSM to load balance the connections across the application hosts.
- We expect database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 1 to DCb without WAAS passed.

Oracle Applications Traffic from Branch 2 to DCa with WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch 2 to DCa.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the ACE load balanced the connections across the application hosts as expected.

The connection from branch to data center was simulated to have T1 bandwidth and 17 ms of latency. The response times were then measured and performance results were quantified. During this test WAAS was accelerating the Oracle traffic.

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 2 to DCa with WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From the ACE, issue the following command in order to verify the ACE's FT status: **show ft group summary** From the ACE, issue the following commands in order to clear the appropriate counters:

```
clear service-policy ORACLE_TCP_TRAFFIC
clear service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS
clear sticky database all
clear serverfarm REDIRECT_PORT_80_TO_PORT_8000
clear serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
clear serverfarm WAE_FARM
```

```
clear stats all
```

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the ACE that you see the connection entries from the LoadRunner generated traffic t o the VIP on ACEby issuing the following command.

```
show conn | inc 10.0..
```

Step 8 Verify the traffic is being accelerated by the WAE devices at the branch by issuing the **show tfo connection summary** command.

From the output of the command you should see all client to application connections established and being fully optimized.

Step 9 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 10 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 11 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 12 Verify on the ACE that you observe the proper expected results by issuing the following commands:

```
show service-policy ORACLE_TCP_TRAFFIC detail
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail
show serverfarm WAE_FARM
show serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP01 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP02 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP03 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show stats http
show stats loadbalance
show stats sticky
show service-policy ORACLE_TCP_TRAFFIC | inc drop
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop
show stats inspect Stop the syslog server and copy the log file to the results file.
```

Step 13 Verify the statistics on the WAE device at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that WAAS will accelerate transaction response times.
- We expect that Oracle transactions submitted through Load Runner will be completed successfully.
- We expect that GSS will direct the Application traffic from Branch to Data Center.
- We expect the ACE to load balance the connections across the application hosts.

Results

Oracle Applications Traffic from Branch 2 to DCa with WAAS passed.

Oracle Applications Traffic from Branch 2 to DCa without WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch2 to DCA.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the ACE load balanced the connections across the application hosts as expected. It was also verified that the Database connections are distributed across both RAC instances

The connection from branch to data center was simulated to have T1(1.5mb/sec) bandwidth and 17ms of latency. The response times were then measured and performance results were quantified.During this test WAAS is disabled

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 2 to DCa without WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From the ACE, issue the following command in order to verify the ACE's FT status: **show ft group summary** From the ACE, issue the following commands in order to clear the appropriate counters:

```
clear service-policy ORACLE_TCP_TRAFFIC
clear service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS
clear sticky database all
clear serverfarm REDIRECT_PORT_80_TO_PORT_8000
clear serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
clear serverfarm WAE_FARM
clear stats all
```

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the ACE that you see the connection entries from the LoadRunner generated traffic t o the VIP on ACE by issuing the following command.

```
show conn | inc 10.0..
```

Step 8 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 9 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 10 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 11 Verify on the ACE that you observe the proper expected results by issuing the following commands:

```
show service-policy ORACLE_TCP_TRAFFIC detail
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail
show serverfarm WAE_FARM
show serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP01 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP02 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP03 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show stats http
show stats loadbalance
show stats sticky
show service-policy ORACLE_TCP_TRAFFIC | inc drop
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop
show stats inspect Stop the syslog server and copy the log file to the results file.
```

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Oracle transactions submitted through Load Runner will be completed successfully.
- We expect transaction response times to be slightly higher since WAAS is disabled.
- We expect the GSS to direct the Application traffic from Branch to Data Center.

- We expect the ACE to load balance the connections across the application hosts.
- We expect the Database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 2 to DCa without WAAS passed.

Oracle Applications Traffic from Branch 2 to DCb with WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch2 to DCB.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the CSM load balanced the connections across the application hosts as expected.It was also verified that the Database connections are distributed across both RAC instances

The connection from branch to data center was simulated to have T1(1.5mb/sec) bandwidth and 19ms of latency. The response times were then measured and performance results were quantified. During this test WAAS was accelerating the Oracle traffic.

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 2 to DCb with WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From the CSM, issue the command, "show mod csm 2 conns vserver wwwin-oefin" and "show mod csm 2 sticky" in order to verify http traffic.

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS, CSM, and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands. On the CSM issue the **clear module csm *module*** connections, **clear module csm *module*** counters, **clear module csm *module*** sticky all commands. On the WAE devices issue the **clear statistics all** command.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal.You should also see the proper dns rule being hit.

Step 7 Verify on the CSM that you see the expected behavior for connections, sticky entries and hits on the real servers by issuing the following commands:

```
show module csm
```

```
module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailFrom these commands you should see an even distribution of connections to each real
server, each with it's own sticky entry based on source IP address.
```

Step 8 Verify the traffic is being accelerated by the WAE devices at the branch by issuing the **show tfo connection summary** command.

From the output of the command you should see all client to application connections established and being fully optimized.

Step 9 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 10 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 11 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 12 Verify on the CSM that you observe the proper expected results by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailCompare this with the pre test output and verify you see the expected changes from
the pre test results.
```

Step 13 Verify the statistics on the WAE device at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
```

```
show statistics dre
```

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that WAAS will accelerate transaction response times.
- We expect transaction response times to be slightly higher in comparision to DCA.
- We expect that Oracle transactions submitted through Load Runner will be completed successfully.
- We expect that GSS will direct the Application traffic from Branch to Data Center.
- We expect that CSM will load balance the connections across the application hosts.
- We expect database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 2 to DCb with WAAS passed.

Oracle Applications Traffic from Branch 2 to DCb without WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch2 to DCB.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the CSM load balanced the connections across the application hosts as expected. It was also verified that the Database connections are distributed across both RAC instances

The connection from branch to data center was simulated to have T1(1.5mb/sec) bandwidth and 19ms of latency. The response times were then measured and performance results were quantified.During this test WAAS is disabled

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 2 to DCb without WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From the CSM, issue the command, "show mod csm 2 conns vserver wwwin-oefin" and "show mod csm 2 sticky" in order to verify http traffic.

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS, CSM, and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands. On the CSM issue the **clear module csm *module*** connections, **clear module csm *module*** counters, **clear module csm *module*** sticky all commands.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the CSM that you see the expected behavior for connections, sticky entries and hits on the real servers by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailFrom these commands you should see an even distribution of connections to each real
server, each with it's own sticky entry based on source IP address.
```

Step 8 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 9 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 10 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 11 Verify on the CSM that you observe the proper expected results by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
```

```
 sticky
show module csm

module
 reals sfarm

sfarm
 detailCompare this with the pre test output and verify you see the expected changes from
the pre test results.
```

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Oracle transactions submitted through Load Runner to be completed successfully.
- We expect transaction response times to be slightly higher since WAAS is disabled.
- We expect the GSS to direct the Application traffic from Branch to Data Center.
- We expect the CSM to load balance the connections across the application hosts.
- We expect database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 2 to DCb without WAAS passed.

Oracle Applications Traffic from Branch 3 to DCa with WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch 3 to DCa.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the ACE load balanced the connections across the application hosts as expected.

The connection from branch to data center was simulated to have T1(1.5mb/sec) bandwidth and 68ms of latency. The response times were then measured and performance results were quantified. During this test WAAS was accelerating the Oracle traffic.

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 3 to DCa with WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from

LR tool. From the ACE, issue the following command in order to verify the ACE's FT status: **show ft group summary** From the ACE, issue the following commands in order to clear the appropriate counters:

```
clear service-policy ORACLE_TCP_TRAFFIC
clear service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS
clear sticky database all
clear serverfarm REDIRECT_PORT_80_TO_PORT_8000
clear serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
clear serverfarm WAE_FARM
clear stats all
```

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the ACE that you see the connection entries from the LoadRunner generated traffic t o the VIP on ACEby issuing the following command.

```
show conn | inc 10.0..
```

Step 8 Verify the traffic is being accelerated by the WAE devices at the branch by issuing the **show tfo connection summary** command.

Step 9 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 10 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 11 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 12 Verify on the ACE that you observe the proper expected results by issuing the following commands:

```
show service-policy ORACLE_TCP_TRAFFIC detail
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail
show serverfarm WAE_FARM
show serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP01 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP02 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP03 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show stats http
show stats loadbalance
show stats sticky
show service-policy ORACLE_TCP_TRAFFIC | inc drop
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop
```

```
show stats inspect Stop the syslog server and copy the log file to the results file.
```

Step 13 Verify the statistics on the WAE device at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that WAAS will accelerate transaction response times.
- We expect that Oracle transactions submitted through Load Runner will be completed successfully.
- We expect that GSS will direct the Application traffic from Branch to Data Center.
- We expect the ACE to load balance the connections across the application hosts.
- We expect database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 3 to DCa with WAAS passed.

Oracle Applications Traffic from Branch 3 to DCa without WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch3 to DCA.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the ACE load balanced the connections across the application hosts as expected. It was also verified that the Database connections are distributed across both RAC instances

The connection from branch to data center was simulated to have T1(1.5mb/sec) bandwidth and 70ms of latency. The response times were then measured and performance results were quantified.During this test WAAS is disabled

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 3 to DCa without WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from

LR tool. From the ACE, issue the following command in order to verify the ACE's FT status: **show ft group summary** From the ACE, issue the following commands in order to clear the appropriate counters:

```
clear service-policy ORACLE_TCP_TRAFFIC
clear service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS
clear sticky database all
clear serverfarm REDIRECT_PORT_80_TO_PORT_8000
clear serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
clear serverfarm WAE_FARM
clear stats all
```

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the ACE that you see the connection entries from the LoadRunner generated traffic t o the VIP on ACE by issuing the following command.

```
show conn | inc 10.0..
```

Step 8 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 9 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 10 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 11 Verify on the ACE that you observe the proper expected results by issuing the following commands:

```
show service-policy ORACLE_TCP_TRAFFIC detail
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS detail
show serverfarm WAE_FARM
show serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP01 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP02 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show sticky database static rserver ORAAPP03 serverfarm ORAAPP_ORACLE_FARM_WAAS_CONTENT
show stats http
show stats loadbalance
show stats sticky
show service-policy ORACLE_TCP_TRAFFIC | inc drop
show service-policy OPTIMIZED_TRAFFIC_TO_ORIGIN_SERVERS | inc drop
show stats inspect Stop the syslog server and copy the log file to the results file.
```

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect Oracle transactions submitted through Load Runner will be completed successfully.
- We expect transaction response times to be much higher due to latency and bandwidth contraints and with WAAS being disabled.
- We expect some transactions to timeout due to high latency.
- We expect GSS will direct the Application traffic from Branch to Data Center.
- We expect ACE will load balance the connections across the application hosts.
- We expect database connections to load balance across the two RAC instances.

Results.

Oracle Applications Traffic from Branch 3 to DCa without WAAS passed.

Oracle Applications Traffic from Branch 3 to DCb with WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch3 to DCB.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the CSM load balanced the connections across the application hosts as expected. It was also verified that the Database connections are distributed across both RAC instances

The connection from branch to data center was simulated to have T1(1.5mb/sec) bandwidth and 70ms of latency. The response times were then measured and performance results were quantified. During this test WAAS was accelerating the Oracle traffic.

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 3 to DCb with WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From the CSM, issue the command, "show mod csm 2 conns vserver wwwin-oefin" and "show mod csm 2 sticky" in order to verify http traffic.

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS, CSM, and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands. On the CSM issue the **clear module csm *module*** connections, **clear module csm *module*** counters, **clear module csm *module*** sticky all commands. On the WAE devices issue the **clear statistics all** command.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the CSM that you see the expected behavior for connections, sticky entries and hits on the real servers by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailFrom these commands you should see an even distribution of connections to each real
server, each with it's own sticky entry based on source IP address.
```

Step 8 Verify the traffic is being accelerated by the WAE devices at the branch by issuing the **show tfo connection summary** command.

From the output of the command you should see all client to application connections established and being fully optimized.

Step 9 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 10 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 11 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 12 Verify on the CSM that you observe the proper expected results by issuing the following commands:

```
show module csm

module
 conns vserver

vserver
```

```
show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailCompare this with the pre test output and verify you see the expected changes from
the pre test results.
```

Step 13 Verify the statistics on the WAE device at the branch by issuing the following commands:

```
show statistics tfo
show statistics tfo saving
show statistics tcp
show statistics dre
```

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that WAAS will accelerate transaction response times.
- We expect transaction response times to be slightly higher in comparision to DCA.
- We expect that Oracle transactions submitted through Load Runner will be completed successfully.
- We expect that GSS will direct the Application traffic from Branch to Data Center.
- We expect that CSM will load balance the connections across the application hosts.
- We expect database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 3 to DCb with WAAS passed.

Oracle Applications Traffic from Branch 3 to DCb without WAAS

This test verified the functionality of the Oracle E-business Applications deployment over the entire network. This involved sending load runner based traffic from Branch3 to DCB.

It was verified that the GSS distribution of client DNS queries worked as expected. Once the GSS has distributed the queries it was verified that the CSM load balanced the connections across the application hosts as expected.It was also verified that the Database connections are distributed across both RAC instances

The connection from branch to data center was simulated to have T1(1.5mb/sec) bandwidth and 70ms of latency. The response times were then measured and performance results were quantified.During this test WAAS is disabled

Test Procedure

The procedure used to perform the Oracle Applications Traffic from Branch 3 to DCb without WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Load Runner(LR) traffic generation tool is set up to send traffic from the Branch host to the Data Center. For each of the defined business flows, the LR tool will source HTML and Oracle Forms applications interaction from simulated clients at the branch to the Application and Database hosts in the data center.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool. From the CSM, issue the command, **show mod csm 2 conns vserver wwwin-oefin** and **show mod csm 2 sticky** in order to verify http traffic.

Step 3 On the GSS verfiy that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 4 Clear the appropriate counters on the GSS, CSM, and WAAS devices in the network. On the GSS issue the **clear statistics keepalive all** and **clear statistics dns** commands. On the CSM issue the **clear module csm *module*** connections, **clear module csm *module*** counters, **clear module csm *module*** sticky all commands.

Step 5 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 6 On the GSS, verify the GSS statitics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

From these commands you should see no dns response errors, dns responses sent and dns responses no error should be equal. You should also see the proper dns rule being hit.

Step 7 Verify on the CSM that you see the expected behavior for connections, sticky entries and hits on the real servers by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailFrom these commands you should see an even distribution of connections to each real
server, each with it's own sticky entry based on source IP address.
```

Step 8 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. First graph will dsiplay the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps to determine percentage of transactions that met predefined response times.

Step 9 Verify the Database connections are being load balanced across both the nodes in RAC. From the output you should see user distribution between both database instances.

Step 10 On the GSS, verify the GSS statisics by issuing the **show statistics dns global** command as well as the **show statistics dns rule** command.

Compare this with the pre-test output and verify you see the expected changes from the pre test output.

Step 11 Verify on the CSM that you observe the proper expected results by issuing the following commands:

```
show module csm

module
 conns vserver

vserver

show module csm

module
 sticky
show module csm

module
 reals sfarm

sfarm
 detailCompare this with the pre test output and verify you see the expected changes from
the pre test results.
```

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect WAAS to accelerate transaction response times.
- We expect Oracle transactions submitted through Load Runner to be completed successfully.
- We expect the GSS to direct the Application traffic from Branch to Data Center.
- We expect the CSM to load balance the connections across the application hosts.
- We expect database connections to load balance across the two RAC instances.

Results

Oracle Applications Traffic from Branch 3 to DCb without WAAS passed.

Oracle E-Business Applications Environment Validation

Oracle E-business Applications is configured in Active/Active hybrid mode where Application Layer is active in both Data Centers and Database is active in one Data Center.This test would validate configuration of the Applications in four categories:

- iAS
- Oracle Applications Framework
- RDBMS

- Environment

Test Procedure

The procedure used to perform the Oracle E-Business Applications Environment Validation test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 This step would validate all the configuration realted to IAS.Verify you can TNSPING and sqlplus the database alias used from APPL_TOP for each of the Application Host.Verify you can tnsping and sqlplus to the database service name alias after sourcing the sid_host.env file in IAS_ORACLE_HOME. Verify you can connect to the database using APPLSYSPUB/PUB account.Verify dbc file in use is valid with right location and permissions

Step 3 Validate web server running on Application hosts and able to render static html.

Step 4 Validate the profile options for the follwing profiles and verify the results.

- APPS_FRAMEWORK_AGENT (Application Framework Agent)
- APPS_JSP_AGENT (Applications JSP Agent)
- APPS_SERVLET_AGENT (Apps Servlet Agent)
- APPS_WEB_AGENT (Applications Web Agent)
- ICX_FORMS_LAUNCHER (ICX: Forms Launcher)
- POR_SERVLET_VIRTUAL_PATH (POR: Servlet Virtual Path)
- GUEST_USER_PWD (Guest User Password)

Step 5 This step will validate the RDBMS setup.Verify the guest user information by running the following sql.

```
select user_name, start_date, end_date
from   fnd_user
where  user_name = 'GUEST';
This should return one row, end_date should be NULL or in advance of today's date, and
start_date should be before today's date.
```

Run the following script to ensure there are no invalid objects:

```
select owner, object_name, object_type
from   all_objects
where  status != 'VALID'
order by owner, object_type, object_name;Validate the FND_NODES table by running the
following sql:

select NODE_NAME, NODE_ID , SERVER_ID , SERVER_ADDRESS from FND_NODES;Validate the
'ICX_SESSIONS_S' synonym Validate the ICX_PARAMETERS table by running the following SQL:

select count(*) from icx_parameters;
```

Step 6 This step will validate access to Oracle Forms through the application.Follow the sequence of steps and verify you can succesfully be able to access forms.

- Login to homepage http://wwwin-oefin.gslb.dcap.com:8000/ and click on Apps Logon Link
- Click on ebusiness home page
- Login using user id: sysadmin and password: sysadmin
- Click System Administrator on Responsibilities navigation pane on the left

- Click Requests under "Concurrent"

Step 7 Validate the concurrent manager setup, submit a batch request and validate the log and report file by viewing the results. Issue the following steps

- Login to homepage http://wwwin-oefin.gslb.dcap.com and click on Apps Logon Link
- Click on ebusiness home page
- Login using user id: sysadmin and password: sysadmin
- Click System Administrator on Responsibilities navigation pane on the left
- Click Requests under "Concurrent"
- Click on Submit new request
- Type Active Resp% and click find
- Select Active Responsibilities and hit submit
- Click View Running requests
- Identify the request you just submitted and wait for it to complete
- View the details

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect TNS connectivity is valid from all the Application hosts.
- We expect sqlnet connections to be load balanced across 2 nodes in the RAC cluster.
- We expect dbc file in use is Valid.
- We expect that web server is running and able to render static html.
- We expect Servlets and JSP are functioning.
- We expect all the profile options values are set appropriately.
- We expect all the Database objects in Valid status.
- We expect all the application nodes are in fnd_nodes table.
- We expect to login successfully into E-biz Application for both forms and HTML modules.

Results

Oracle E-Business Applications Environment Validation passed.

CHAPTER 15

Microsoft Exchange

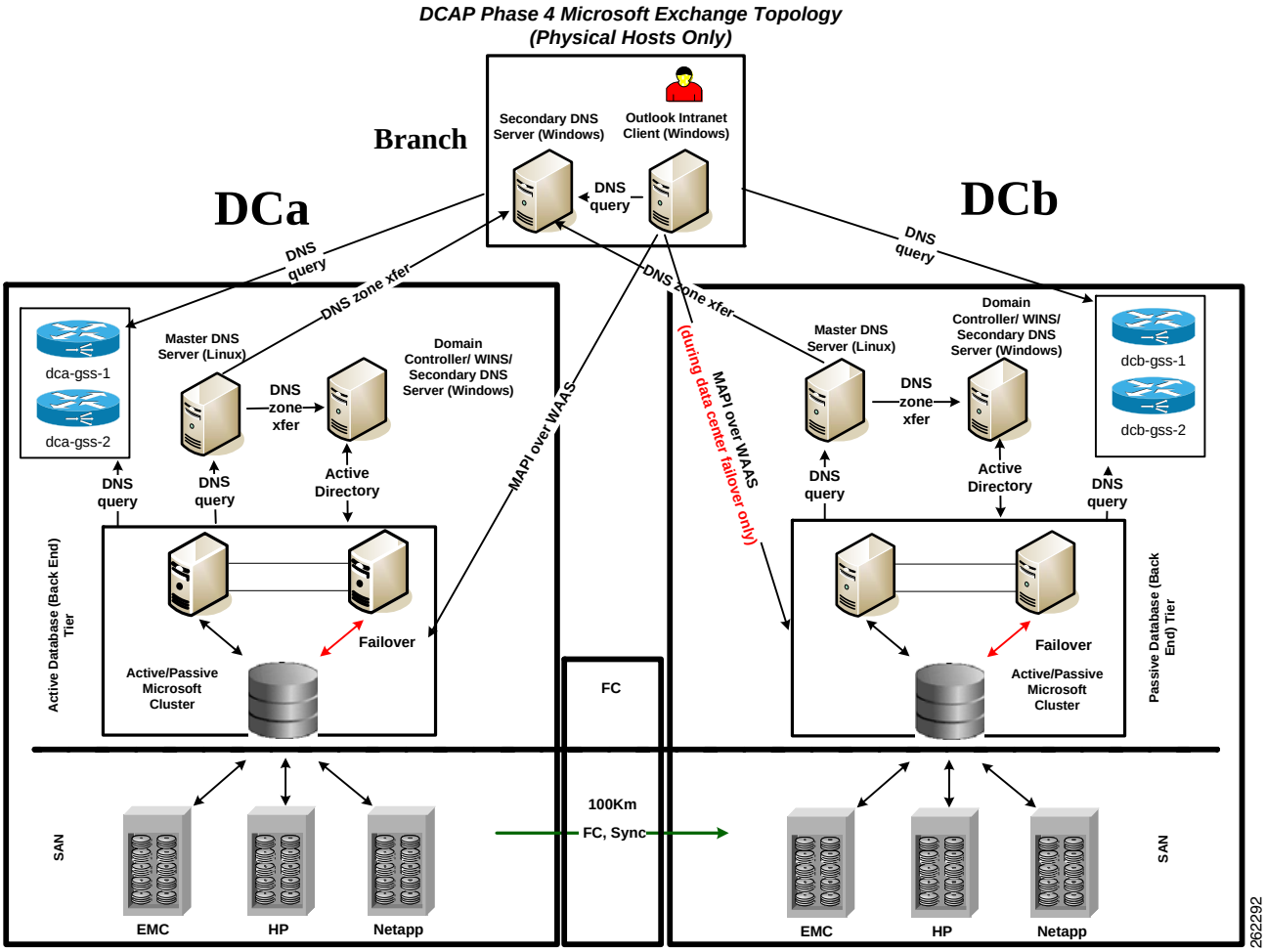
DCAP phase 4 testing includes Microsoft Exchange 2003 and Outlook 2003.

MS Exchange 2003 Topology

The topology consists of two Windows 2003 active/passive back end clusters, one in each data center. The primary cluster hosts an Exchange Virtual Server called "DCAP-MBOX-1" and the other cluster acts as a disaster recovery/business continuance standby cluster. The clusters use fibre channel to attach to storage from EMC, HP, and Network Appliance. This storage is replicated synchronously from the primary to the standby cluster. Tests include running Microsoft Jetstress on the primary cluster, failing the primary cluster over the the standby cluster, and failing the standby cluster back to the primary cluster. Client access for failover/failback testing is from Outlook 2003 clients at three remote branches via the MAPI protocol over the test intranet, which is accelerated by WAAS. DNS lookup services were provided by GSS to allow proper site selection during failover/failback. The DCAP roadmap includes Exchange 2007, asynchronous replication, and other features such as front end servers, server load balancing, firewall, SSL offload, and proxy.

Figure 15-1 depicts the primary Microsoft Exchange 2003 application components and some basic data flow information to how they relate to each other.

Figure 15-1 DCAP Exchange Test Topology



The components include the following:

- A primary active/passive Microsoft Windows cluster in DCa and a similarly configured failover active/passive cluster in DCb. These clusters provide a high-availability environment for the Microsoft Exchange 2003 virtual server called "DCAP-MBOX-1" that is used throughout the tests. These clusters are separate and self-contained; that is, the hosts are not configured as part of a geographically dispersed cluster.
- SAN storage is connected and replicated through the DCAP SAN testbed in both data centers. The Exchange data is located on SAN storage that's synchronously replicated over a simulated 100 km distance from DCa to DCb. Fibre channel-attached storage from three different vendors, EMC, Hewlett Packard, and NetApp, is used, and the replication mechanism is SRDF/S, Continuous Access XP Synchronous, and synchronous SnapMirror, respectively.
- A total of four GSS appliances, two per data center, providing name service for the dcap-mbox-1.gslb.dcap.com FQDN.
- Master DNS Linux servers, one per data center, where manual administrator updates occur.
- Secondary DNS Windows servers, one per data center and one per branch, which are automatically updated through zone file transfers. Branch client hosts use the local secondary DNS server for queries.

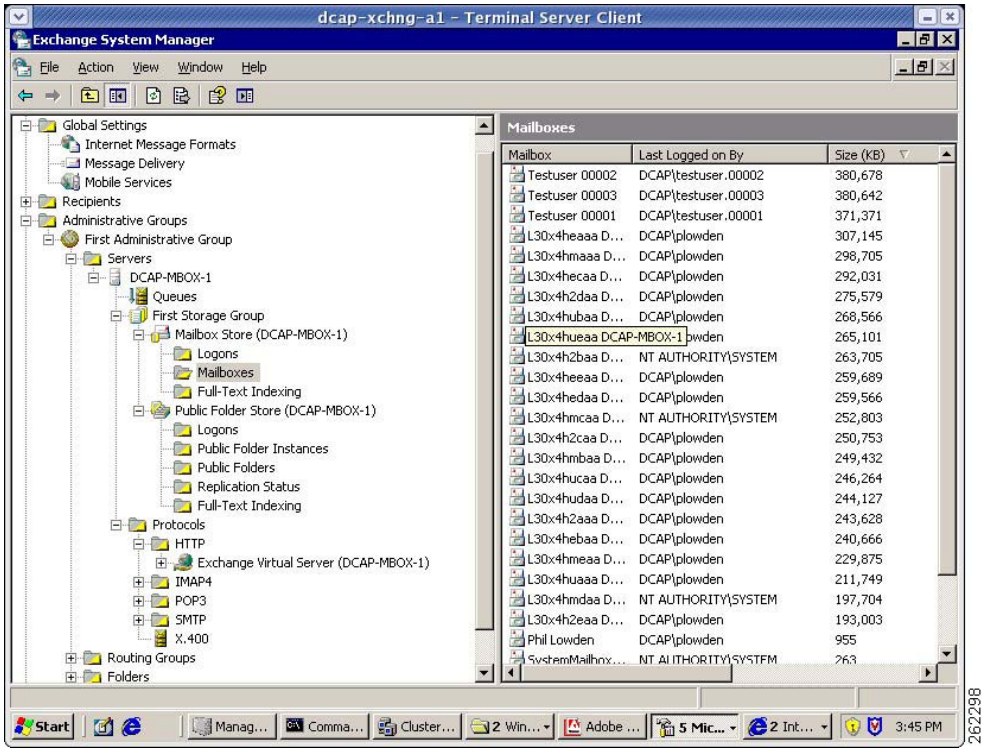
- Windows Domain Controller servers, one per data center. The Windows domain is called "dcap.com" (or DCAP for pre-Windows 2000 hosts).
- Microsoft Outlook 2003 clients, one for each of the three branches (only one branch is shown). The Outlook clients access the back-end Exchange server using the MAPI protocol over the DCAP testbed WAN infrastructure, which incorporates WAAS to optimize MAPI traffic.

For more details about the SAN, GSS, WAN, WAAS, and LAN topologies used by Exchange, refer to the corresponding sections in the DCAP documentation.

The Exchange implementation incorporated best practice information from Microsoft and third-party vendors (for example: http://technet.microsoft.com/en-us/library/0c968830-aaba-4938-9115-85d2a09736e4.aspx).

In this phase, a very basic Exchange environment, consisting of a single Exchange Virtual Server running on a back-end Microsoft Windows cluster, is used. This server uses storage in one database, and only one storage group is used. Microsoft Exchange System Manager was used to manage the environment. Figure 15-2 shows a screen dump of Exchange System Manager.

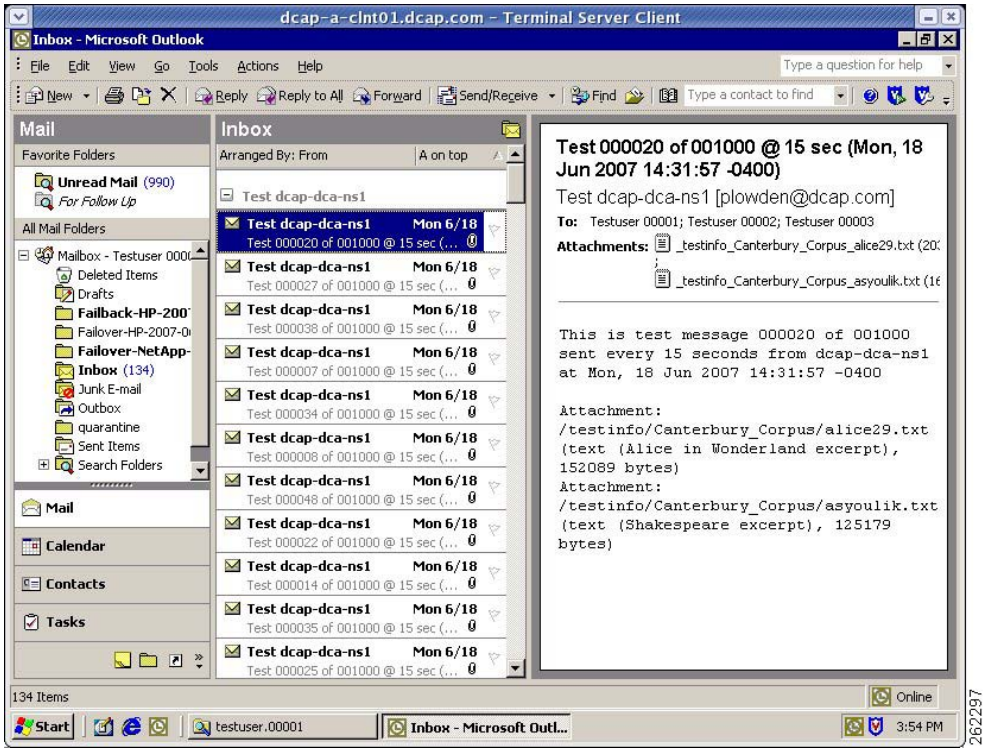
Figure 15-2 DCAP MS Exchange 2003 Exchange System Manager Screen Capture



Notice the DCAP-MBOX-1 virtual server on the left. On the right is the list of mailboxes, which constitutes the primary data handled by Exchange.

Clients are all Outlook 2003 clients using MAPI to access Exchange over the DCAP intranet. No front-end Exchange servers are used. Figure 15-3 shows a screen dump of Outlook 2003.

Figure 15-3 Outlook 2003 Screen Capture



From a storage perspective, the DCAP SAN testbed and storage frames from EMC, HP, and NetApp, are used. The storage is configured similarly for each vendor. At any given time the Exchange hosts accessed only one vendor. This is both to approximate a real Exchange environment and to ensure each storage vendor's preferred revision levels and multipathing software was used. For EMC, PowerPath provided multipath support. For HP, multipathing was through HP MPIO Full Featured DSM for XP Disk Arrays. For NetApp, the multipathing software was ONTAP DSM 3.0.

For all vendors, the storage layout is as follows:

- Drive E: Exchange database files (.edb, .stm)
- Drive F: Exchange SMTP queue files
- Drive G: Exchange log files (.log)

Each cluster also had a cluster quorum disk to enable Microsoft Cluster Server (MSCS) to be used to provide high-availability for Exchange in each data center.

Note The MSCS clusters in each data center are separate. In this phase, geographically dispersed clustering is not used.

Figure 15-4 shows a screen dump of Cluster Administrator from the primary cluster node depicting the cluster resource group.

Figure 15-4 Cluster Administrator Cluster Resource Group Screen Capture

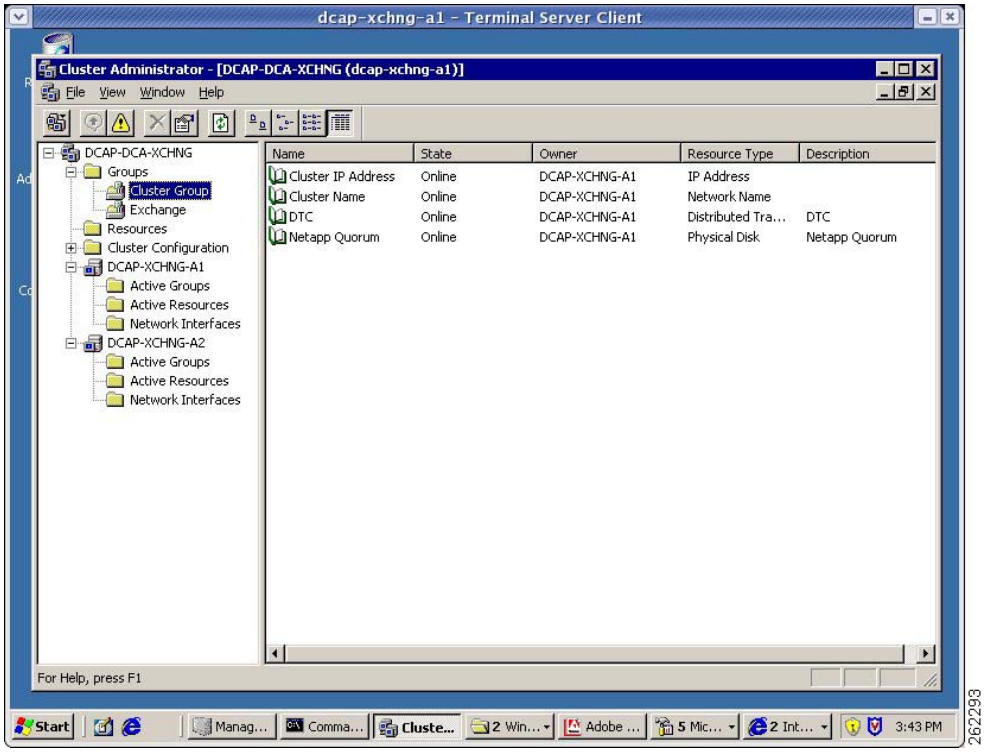
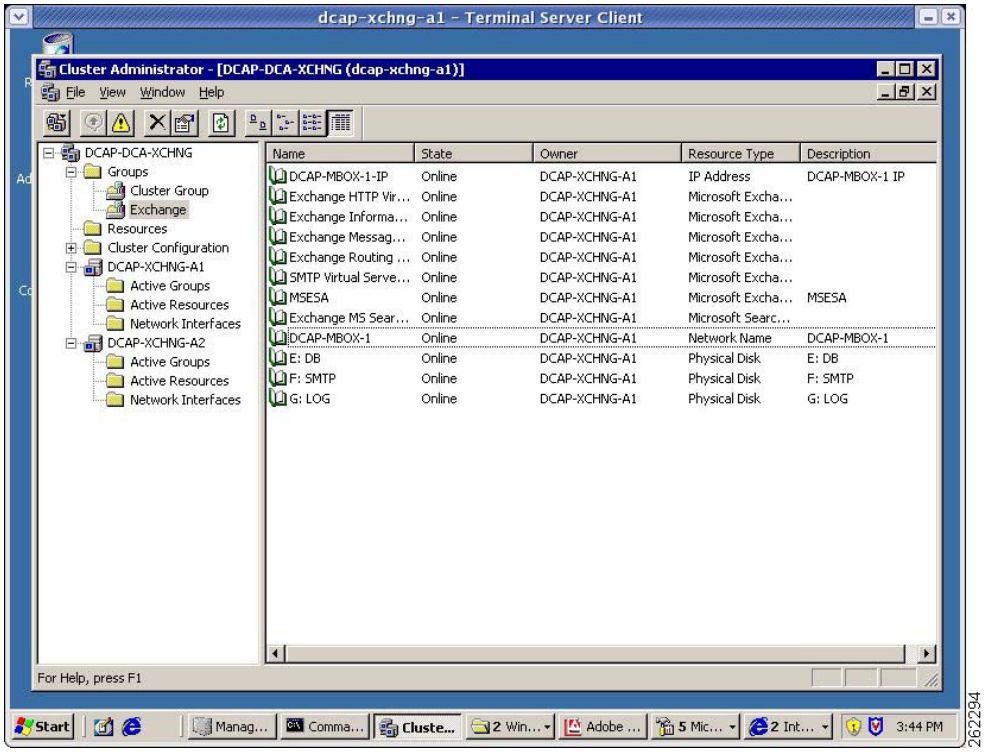


Figure 15-5 shows a screen dump of Cluster Administrator from the primary cluster node depicting the Exchange resource group.

Figure 15-5 Cluster Administrator Cluster Resource Group Screen Capture



The Cisco DCAP 4.0 Exchange tests fall into two major categories: Fabric Extension and Disaster Recovery.

The Fabric Extension tests checked synchronous storage replication of Microsoft Exchange 2003 data over the SAN topology. A simulated distance of 100 km was in effect for all tests. For each storage vendor, synchronous tests over FC are performed with and without FC write acceleration enabled on the transit fabric. For EMC, the replication mechanism was SRDF/S. For HP, it was Continuous Access XP Synchronous. For NetApp, it was synchronous SnapMirror.

A Microsoft tool facilitated Fabric Extension testing, namely Jetstress for simulating Exchange disk I/O load. For more information on Jetstress, refer to http://go.microsoft.com/fwlink/?linkid=27883.

Jetstress was used to generate a lot of I/O to the SAN storage while the SAN replication path was configured without FC write acceleration. (The tool is called "Jetstress" because the Exchange database is also called the "Jet database.") Although the Jetstress Disk Performance Test was configured to run for two hours, the actual time the tests took was five to six hours due to the creation of database files and gathering of baseline performance information. The results from this test were compared with results when FC write acceleration was operational. There was some evidence of the expected improvement in throughput in the Jetstress tests. Table 15-1 summarizes these results.

Table 15-1 Jetstress Test Results

Vendor	Without FCWA		With FCWA		Write Speedup
	Write/Sec	Total OPS	Write/Sec	Total IOPS	
EMC	293	334	371	393	27%
HP	423	567	489	638	16%
NetApp	402	511	721	981	79% (NOTE)

Note This result is unexpected, since NetApp uses the FC-VI protocol which is not accelerated by FCWA. This result is due to Exchange running out of memory and doing more disk I/O than normal as a result.

Please see the detailed test results for more information.

Figure 15-6 shows a screen dump of Jetstress from the primary cluster node after configuration and before execution of a test.

Figure 15-6 Jetstress Screen Capture

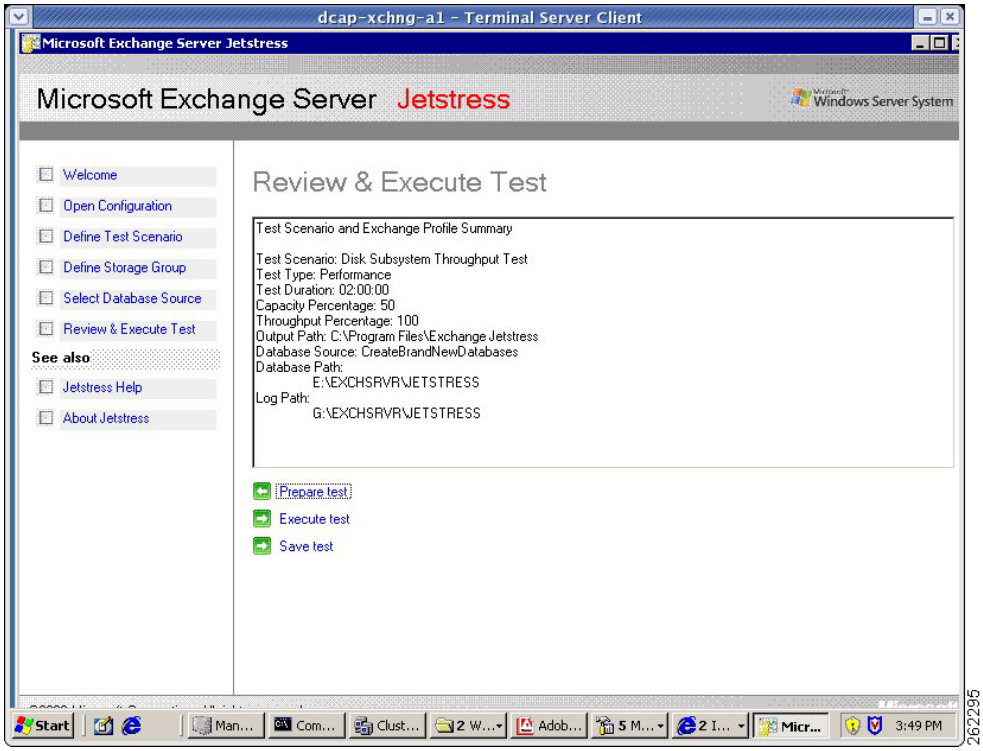
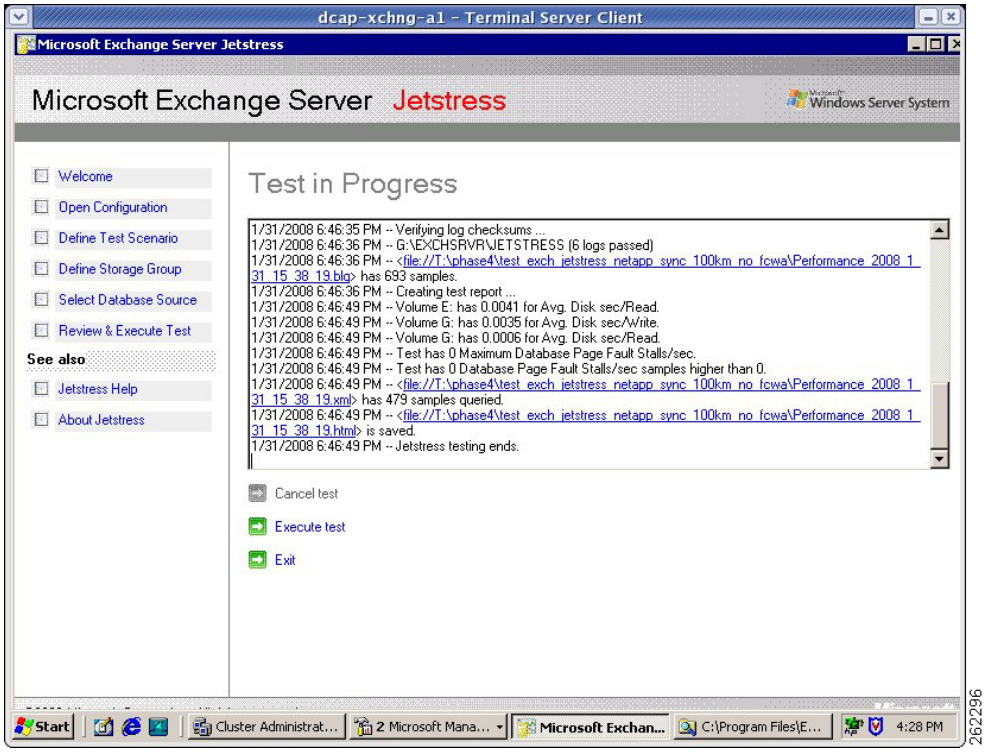


Figure 15-7 shows a screen dump of a Jetstress HTML report from the primary cluster node after execution of a test.

Figure 15-7 Jetstress Screen Capture



Although Windows produces binary output files from Performance Manager which are linked in the Jetstress reports, these files are not included in this document due to their large size (80 MB and larger).

The Disaster Recovery tests ensured synchronous storage replication of Microsoft Exchange 2003 data over the SAN topology was properly configured to support failover of the application to DCb and failback to DCa. A simulated distance of 100 km was in effect for all tests. For each vendor, synchronous tests over FC were performed with fibre channel write acceleration enabled on the transit fabric.

The key metrics used were *recovery point objective*, or RPO, which is the amount of data that could not be failed over, and *recovery time objective*, or RTO, which is the time it took to restore application access at the DCb. Because replication used the synchronous mode, RPO for all vendors was expected to be 0. RTO was dependent on the particular process needed by each storage vendor to failover and failback the storage as well as the manual process for bringing up Exchange. This process did not including updating DNS at the branches to allow clients to adjust to the change in the IP address, since GSS handled this automatically.

Note Because the two data centers have no L2 adjacencies in this phase as a deliberate design goal, using the same IP address for the Exchange Virtual Server in both data centers was not an option.

The primary Microsoft references for how to do the the failover and failback of Exchange itself included

How to Move All Exchange Virtual Servers from a Production Exchange 2003 Cluster to a Standby Exchange 2003 Cluster: (http://technet.microsoft.com/en-us/library/aa996470.aspx)

and

http://www.microsoft.com/technet/prodtechnol/exchange/guides/DROpsGuide/f4d7aa56-abed-4645-b2f8-952191d1c050.mspx.

For failover, three separate tests, one for each storage vendor, were performed. The tests approximate what a practice failover test might look like in an actual Exchange environment. The tests consisted of sending email continuously every 15 seconds from two different hosts to three separate Outlook email recipients, one per branch. The emails contained a short body and two attachments. Then the Exchange stores were dismounted to simulate a storage issue and the SAN storage was failedover to DCb and Outlook was brought back online on the failover Exchange cluster. No network disruption occurred, since the purpose of this test was to make sure storage failover works properly. After the failover, the branch Outlook clients were checked to make sure that all emails sent prior to the simulated failure of the primary environment in DCa were received by clients once Exchange was online in DCb.

For failback, three separate tests, one for each storage vendor, were performed. The tests approximate what a practice failback test might look like in an actual Exchange environment. The tests are similar to the failover tests, except at the start of the test Exchange was running in DCb. By the end of the test, Exchange was once again running on the primary cluster in DCa and Outlook clients at each branch were checked to make sure that no email was lost.

Table 15-2 summarizes the results of failover and failback testing.

Table 15-2 Failover and Failback Test Results

	Failover		**Failback**	
Vendor	**RPO**	**RTO**	**RPO**	**RTO**
EMC	0	8 min	0	8 min
HP	0	39 min (NOTE)	0	6 min
NetApp	0	6 min	0	7 min

Note HP failover RTO was abnormally high due to need to reboot failover cluster to get it to access storage after SAN failover.

For details about how to perform the failover and failback, refer to Volume 11: Data Center High Availability.

Test Results Summary

Table 15-3 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 15-3 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTS's.

Table 15-3 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Disaster Recovery, page 15-11	Fail Over, page 15-11	1. Exchange EMC Fail Over Sync (100km Write Acceleration) 2. Exchange HP Fail Back Sync (100km Write Acceleration) 3. Exchange NetApp Fail Back Sync (100km Write Acceleration)	
Disaster Recovery, page 15-11	Fail Back, page 15-15	1. Exchange EMC Fail Back Sync (100km Write Acceleration) 2. Exchange HP Fail Back Sync (100km Write Acceleration) 3. Exchange NetApp Fail Back Sync (100km Write Acceleration)	
Fabric Extension, page 15-18	EMC, page 15-19	1. Jetstress with EMC Sync Replication (100km no FC Write Acceleration) 2. Jetstress with EMC Sync Replication (100km with FC Write Acceleration)	
Fabric Extension, page 15-18	HP, page 15-21	1. Jetstress HP Sync (100km FC Write Acceleration) 1. Jetstress HP Sync (100km FC Write Acceleration)	
Fabric Extension, page 15-18	NetApp, page 15-23	1. Jetstress NetApp Sync (100km FC Write Acceleration) 2. Jetstress NetApp Sync (100km FC Write Acceleration)	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Disaster Recovery

The disaster recovery tests ensure synchronous storage replication of Microsoft Exchange 2003 data over the SAN topology is properly configured to support failover of the application to data center B and failback to data center A. A simulated distance of 100 km was in effect for all tests. For each vendor, synchronous tests over FC are performed with fibre channel write acceleration enabled on the transit fabric. GSS provided site selection services for clients in three remote branch locations.

This section contains the following topics:

Fail Over

Three separate tests, one for each storage vendor, are performed. The tests approximate what a practice fail over test might look like in an actual Exchange environment. The tests consist of sending email continuously every 15 seconds from two different hosts to three separate Outlook email recipients, one per branch. The emails contain a short body and two attachments. Then the Exchange stores are dismounted to simulate a storage issue and the SAN storage is failed over to data center B and Outlook is brought back online on the failover Exchange cluster. No network disruption occurs, since the purpose of this test is to make sure storage fail over works properly. After the fail over, the branch Outlook clients are checked to make sure that all emails sent prior to the simulated failure of the primary environment in data center A are received by clients once Exchange is online in data center B.

This section contains the following topics:

Exchange EMC Fail Over Sync (100km Write Acceleration)

This test ensures a Microsoft Exchange database that's replicated to a remote data center can be failed over for disaster recovery/business continuance purposes. In this test, the Exchange database devices are on a SAN-attached storage frame that is synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is EMC SRDF/S. FC write acceleration is enabled on the replication MDS switches. This test is not meant to simulate a disaster; it is meant to verify the configurations and procedures are in place to enable a fail over in the event of a disaster. In other words, this test is a practice fail over.

Test Procedure

The procedure used to perform the Exchange EMC Fail Over Sync (100km Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify basic conditions (FC Write Acceleration is enabled, Exchange is running on the primary node in the primary cluster, the appropriate latency is set, and replication is operating correctly). Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Send email messages continuously throughout the test from one server in each data center to each of three test Outlook users, one at each branch location.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that an Exchange fail over, complete with DNS changes, will succeed in about 30 to 60 minutes with no loss of data.
- We expect no CPU or memory problems.

Results

Exchange EMC Fail Over Sync (100km Write Acceleration) passed.

Exchange HP Fail Over Sync (100km Write Acceleration)

This test ensures a Microsoft Exchange database that's replicated to a remote data center can be failed over for disaster recovery/business continuance purposes. In this test, the Exchange database devices are on a SAN-attached storage frame that is synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is HP Continuous Access XP Sync. FC write acceleration is enabled on the replication MDS switches. This test is not meant to simulate a disaster; it is meant to verify the configurations and procedures are in place to enable a fail over in the event of a disaster. In other words, this test is a practice fail over.

Test Procedure

The procedure used to perform the Exchange HP Fail Over Sync (100km Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify basic conditions (FC Write Acceleration is enabled, Exchange is running on the primary node in the primary cluster, the appropriate latency is set, and replication is operating correctly). Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Send email messages continuously throughout the test from one server in each data center to each of three test Outlook users, one at each branch location.

Step 4 Gracefully fail over Exchange, including the storage and cluster:

1. Using Exchange System manager, dismount the Public Folder and Mailbox stores.
2. Using Cluster Administrator, offline the Exchange Virtual Server resource group on the primary Exchange cluster (do *not* remove the Exchange Virtual Server). Then offline the Exchange service group.
3. Using the HORCM CLI on dcap-san-hst-10 on the primary frame, delete the synchronous CA pairs with a "pairsplit" command (without the force option enabled). Rescan disks as needed using Disk Manager on the primary fail over cluster node, dcap-xchng-b1.
4. On a domain controller, reset the domain account for the Exchange Virtual Server.
5. On the primary fail over cluster node, create as needed and online the Exchange service group in Cluster Administrator; be sure to enable Kerberos and uncheck the "ensure DNS changes succeed box" in the network name resource before onlining.
6. On the primary fail over cluster node, create a Microsoft Exchange System Attendant resource. Using Cluster Administrator on the primary fail over cluster node, online the Exchange service group.
7. Then in Exchange System Manager, mount both the mailbox and public folders stores as needed, and ensure the HTTP and SMTP protocols are using the data center B address (Properties).
8. Verify DNS points to the correct address on all three branch DNS secondary servers.
9. Ensure the branch Outlook users begin to receive email again. Check for email until the queues are clear on both sending hosts. Flush the email queues on the hosts sending the emails as needed.
10. Verify email reception.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that an Exchange fail over, complete with DNS changes, will succeed in about 30 to 60 minutes with no loss of data.
- We expect no CPU or memory problems.

Results

Exchange HP Fail Over Sync (100km Write Acceleration) passed.

Exchange NetApp Fail Over Sync (100km Write Acceleration)

This test ensures a Microsoft Exchange database that's replicated to a remote data center can be failed over for disaster recovery/business continuance purposes. In this test, the Exchange database devices are on a SAN-attached storage frame that is synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is NetApp synchronous SnapMirror. FC write acceleration is enabled on the replication MDS

switches. This test is not meant to simulate a disaster; it is meant to verify the configurations and procedures are in place to enable a fail over in the event of a disaster. In other words, this test is a practice fail over.

Test Procedure

The procedure used to perform the Exchange NetApp Fail Over Sync (100km Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify basic conditions (FC Write Acceleration is enabled, Exchange is running on the primary node in the primary cluster, the appropriate latency is set, and replication is operating correctly). Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Send email messages continuously throughout the test from one server in each data center to each of three test Outlook users, one at each branch location.

Step 4 Gracefully fail over Exchange, including the storage and cluster:

1. Using Exchange System manager, dismount the Public Folder and Mailbox stores.
2. Using Cluster Administrator, offline the Exchange Virtual Server resource group on the primary Exchange cluster (do *not* remove the Exchange Virtual Server). Then offline the Exchange service group.
3. Using the CLI on the primary filer, quiesce and break each snapmirror relationship. Rescan disks as needed using Disk Manager on the primary fail over cluster node, dcap-xchng-b1.
4. On a domain controller, reset the domain account for the Exchange Virtual Server.
5. On the primary fail over cluster node, create as needed and online the Exchange service group in Cluster Administrator; be sure to enable Kerberos and uncheck the "ensure DNS changes succeed box" in the network name resource before onlining.
6. On the primary fail over cluster node, create a Microsoft Exchange System Attendant resource. Using Cluster Administrator on the primary fail over cluster node, online the Exchange service group.
7. Then in Exchange System Manager, mount both the mailbox and public folders stores as needed, and ensure the HTTP and SMTP protocols are using the data center B address (Properties).
8. Verify DNS points to the correct address on all three branch DNS secondary servers.
9. Ensure the branch Outlook users begin to receive email again. Check for email until the queues are clear on both sending hosts. Flush the email queues on the hosts sending the emails as needed.
10. Verify email reception.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that an Exchange fail over, complete with DNS changes, will succeed in about 30 to 60 minutes with no loss of data.

- We expect no CPU or memory problems.

Results

Exchange NetApp Fail Over Sync (100km Write Acceleration) passed.

Fail Back

Three separate tests, one for each storage vendor, are performed. The tests approximate what a practice fail back test might look like in an actual Exchange environment. The tests are similar to the fail over tests, except at the start of the test Exchange is running in data center B. By the end of the test, Exchange is once again running on the primary cluster in data center A and Outlook clients at each branch are checked to make sure that no email was lost.

This section contains the following topics:

Exchange EMC Fail Back Sync (100km Write Acceleration)

This test ensures a Microsoft Exchange database that's failed over to a remote data center for disaster recovery/business continuance purposes can be failed back to the original data center. In this test, the Exchange database devices are on a SAN-attached storage frame that prior to the fail over had been synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is EMC SRDF/S. FC write acceleration is enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the Exchange EMC Fail Back Sync (100km Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify basic conditions (FC Write Acceleration is enabled, the correct latency is applied, Exchange is running on the primary node in the primary cluster, and both the cluster and storage frame in the primary data center are in the correct state for failback). Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Just prior to the start of the outage window, send email messages continuously throughout the test from one server in each data center to each of three test Outlook users, one at each branch location.

Step 4 Stop background scripts to collect final status of network devices and analyze for error.

Step 5 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that an Exchange fail back, complete with DNS changes, will succeed in about 30 to 60 minutes with no loss of data.
- We expect no CPU or memory problems.

Results

Exchange EMC Fail Back Sync (100km Write Acceleration) passed.

Exchange HP Fail Back Sync (100km Write Acceleration)

This test ensures a Microsoft Exchange database that's failed over to a remote data center for disaster recovery/business continuance purposes can be failed back to the original data center. In this test, the Exchange database devices are on a SAN-attached storage frame that prior to the fail over had been synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is HP Continuous Access XP Sync. FC write acceleration is enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the Exchange HP Fail Back Sync (100km Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify basic conditions (FC Write Acceleration is enabled, the correct latency is applied, Exchange is running on the primary node in the failover cluster, and both the cluster and storage in the primary data center are in the correct state for failback). Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Re-establish the SAN replication link(s) as needed and begin resyncing data from the failover data center to the primary data center.

Step 4 Just prior to the outage window, send email messages continuously throughout the test from one server in each data center to each of three test Outlook users, one at each branch location.

Step 5 Follow these steps:

```
Once the pairs established in step 2 are in PAIR status and the outage window has started,
gracefully fail back Exchange, including the storage and cluster:
1. Using Exchange System Manager, dismount the mailbox and public folders stores on the
primary fail over cluster node.
2. Using Cluster Administrator, offline the Exchange service group in the fail over
Exchange cluster and delete the Microsoft Exchange System Attendant resource (do *not*
remove the Exchange Virtual Server).
3. Using the HORCM CLI, fail back and switch the replication direction for each device
pair with a "pairresync -I0 -swaps -g dcap-dca-xchng-s" command on the primary frame.
4. On the primary fail over cluster node, create as needed and online the Exchange service
group in Cluster Administrator; be sure to disable Kerberos and uncheck the "ensure DNS
changes succeed box" in the network name resource before onlining.   (NOTE: this assumes
the Exchange System Attendant Resource was deleted earlier; if not, skip steps 4 and 6.)
5. On a domain controller, reset the domain account for the Exchange Virtual Server.
6. Back on the primary cluster node, create a Microsoft Exchange System Attendant
resource.  Be sure to enable Kerberos on the network name resource and check the "ensure
DNS changes succeed box".
```

```
7. Using Cluster Administrator, online the Exchange service group in the primary Exchange
cluster. Then, using Exchange System Manager, mount both the mailbox and public folders
stores as needed, and verify the HTTP and SMTP protocols are using the proper address.
8. Verify DNS points to the correct address on all three branch DNS secondary servers;
manually reload zone files and purge DNS cache as needed.
9. Ensure the branch Outlook users begin to receive email again.  Check for email until
the queues are clear on both sending hosts.
10. Verify no email is lost.
```

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that an Exchange fail back, complete with DNS changes, will succeed in about 30 to 60 minutes with no loss of data. (NOTE: this time does not include the time required to resync data from the failover data center back to the primary data center.)
- We expect no CPU or memory problems.

Results

Exchange HP Fail Back Sync (100km Write Acceleration) passed.

Exchange NetApp Fail Back Sync (100km Write Acceleration)

This test ensures a Microsoft Exchange database that's failed over to a remote data center for disaster recovery/business continuance purposes can be failed back to the original data center. In this test, the Exchange database devices are on a SAN-attached storage frame that prior to the fail over had been synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is NetApp synchronous SnapMirror. FC write acceleration is enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the Exchange NetApp Fail Back Sync (100km Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify basic conditions (FC Write Acceleration is enabled, the correct latency is applied, Exchange is running on the primary node in the failover cluster, and both the cluster and storage in the primary data center are in the correct state for failback). Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Just prior to the outage window, send email messages continuously throughout the test from one server in each data center to each of three test Outlook users, one at each branch location.

Step 4 Follow these steps:

```
Once the outage window has started, gracefully fail back Exchange, including the storage
and cluster:
```

```
1. Using Exchange System Manager, dismount the mailbox and public folders stores on the
primary fail over cluster node.
2. Using Cluster Administrator, offline the Exchange service group in the fail over
Exchange cluster and delete the Microsoft Exchange System Attendant resource (do *not*
remove the Exchange Virtual Server).
3. Using the CLI, reverse snapmirror for each device.
4. On the primary fail over cluster node, create as needed and online the Exchange service
group in Cluster Administrator; be sure to disable Kerberos and uncheck the "ensure DNS
changes succeed box" in the network name resource before onlining.   (NOTE: this assumes
the Exchange System Attendant Resource was deleted earlier; if not, skip steps 4 and 6.)
5. On a domain controller, reset the domain account for the Exchange Virtual Server.
6. Back on the primary cluster node, create a Microsoft Exchange System Attendant
resource.  Be sure to enable Kerberos on the network name resource and check the "ensure
DNS changes succeed box".
7. Using Cluster Administrator, online the Exchange service group in the primary Exchange
cluster. Then, using Exchange System Manager, mount both the mailbox and public folders
stores as needed, and verify the HTTP and SMTP protocols are using the proper address.
8. Verify DNS points to the correct address on all three branch DNS secondary servers.
9. Ensure the branch Outlook users begin to receive email again.  Check for email until
the queues are clear on both sending hosts.
10. Verify no email is lost.
```

Step 5 Reestablish replication from the primary to the failover data center.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that an Exchange fail back, complete with DNS changes, will succeed in about 30 to 60 minutes with no loss of data. (NOTE: this time does not include the time required to resync data from the failover data center back to the primary data center.)
- We expect no CPU or memory problems.

Results

Exchange NetApp Fail Back Sync (100km Write Acceleration) passed.

Fabric Extension

The fabric extension tests check synchronous storage replication of Microsoft Exchange 2003 data over the SAN topology. A simulated distance of 100 km was in effect for all tests. For each vendor, synchronous tests over FC are performed with and without FC write acceleration enabled on the transit fabric. Microsoft's Jetstress tool is used.

This section contains the following topics:

EMC

The synchronous replication tests for EMC tests SRDF/S over FC with and without FC write acceleration.

This section contains the following topics:

Jetstress with EMC Sync Replication (100km no FC Write Acceleration)

This test runs Microsoft's 2-hour Jetstress Disk Performance Test utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. Jetstress verifies the performance of the disk subsystem as well as the SAN connectivity and replication infrastructure. Jetstress produces an HTML report with such information as overall and per-disk I/Os per second and throughput. Part of the report is very large binary performance log (.blg) files that show performance at a very granular level of detail. These files are not capatured as part of this test, but the basic HTML report is. For more information on Jetstress, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27883. Jetstress uses only the database and log devices. In this test, these devices are on a SAN-attached storage frame that is synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is EMC SRDF. FC write acceleration is not enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the Jetstress with EMC Sync Replication (100km no FC Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify FC Write Acceleration is not enabled. Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Run Jetstress on primary Exchange cluster node after verifying through Windows Cluster Administrator that the Cluster and Exchange groups are running there and the database and log devices are on the proper type of replicated storage. Chose 50% for capacity, 100% for throughput, and 2 hours for duration. Use the database backup in F:\EXCHSRVR\JETSTRESS as the source of the database if it contains a Jetstress.bak file, otherwise choose to create a new database. Also be sure to supress automatic tuning and use 4 threads per storage group. The test type is performance (rather than streaming backup or soft recovery).

Step 4 After Jetstress is done, check HTML report, switch counters, host status, and replication status and verify expected behavior.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that Jetstress will finish successfully in 2 hours and show throughput and I/O per second numbers without FC Write Acceleration that can be compared with corresponding numbers from another test with FC Write Acceleration enabled.
- We expect no CPU or memory problems.

Results

Jetstress with EMC Sync Replication (100km no FC Write Acceleration) passed.

Jetstress with EMC Sync Replication (100km with FC Write Acceleration)

This test runs Microsoft's 1-hour Jetstress Disk Performance Test utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. Jetstress verifies the performance of the disk subsystem as well as the SAN connectivity and replication infrastructure. Jetstress produces an HTML report with such information as overall and per-disk I/Os per second and throughput. Part of the report is very large binary performance log (.blg) files that show performance at a very granular level of detail. These files are not captured as part of this test, but the basic HTML report is. For more information on Jetstress, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27883. Jetstress uses only the database and log devices. In this test, these devices are on a SAN-attached storage frame that is synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is EMC SRDF. FC write acceleration is enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the Jetstress with EMC Sync Replication (100km with FC Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify FC Write Acceleration is enabled. Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Run Jetstress on primary Exchange cluster node after verifying through Windows Cluster Administrator that the Cluster and Exchange groups are running there and the database and log devices are on the proper type of replicated storage. Chose 50% for capacity, 100% for throughput, and 2 hours for duration. Use the database backup in F:\EXCHSRVR\JETSTRESS as the source of the database if it contains a Jetstress.bak file, otherwise choose to create a new database. Also be sure to supress automatic tuning and use 4 threads per storage group. The test type is performance (rather than streaming backup or soft recovery).

Step 4 After Jetstress is done, check HTML report, switch counters, host status, and replication status and verify expected behavior.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that Jetstress will finish successfully in 2 hours and show more throughput and I/Os per second with FC Write Acceleration enabled than without it.
- We expect no CPU or memory problems.

Results

Jetstress with EMC Sync Replication (100km with FC Write Acceleration) passed.

HP

The synchronous replication test for HP tests HP Continuous Access XP Synchronous replication with and without FC write acceleration.

This section contains the following topics:

- Jetstress HP Sync (100km FC Write Acceleration), page 15-21
- Jetstress HP Sync (100km FC Write Acceleration), page 15-22

Jetstress HP Sync (100km FC Write Acceleration)

This test runs Microsoft's 2-hour Jetstress Disk Performance Test utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. Jetstress verifies the performance of the disk subsystem as well as the SAN connectivity and replication infrastructure. Jetstress produces an HTML report with such information as overall and per-disk I/Os per second and throughput. Part of the report is very large binary performance log (.blg) files that show performance at a very granular level of detail. These files are not capatured as part of this test, but the basic HTML report is. For more information on Jetstress, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27883. Jetstress uses only the database and log devices. In this test, these devices are on a SAN-attached storage frame that is synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is Continous Access XP Sync. FC write acceleration is enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the Jetstress HP Sync (100km FC Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify FC Write Acceleration is enabled. Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Run Jetstress on primary Exchange cluster node after verifying through Windows Cluster Administrator that the Cluster and Exchange groups are running there and the database and log devices are on the proper type of replicated storage. Choose 50% for capacity, 100% for throughput, and 2 hours for duration. The test type is performance (rather than streaming backup or soft recovery).

Step 4 After Jetstress is done, check HTML report, switch counters, host status, and replication status and verify expected behavior.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that Jetstress will finish successfully in 2 hours and show more throughput and I/Os per second with FC Write Acceleration enabled than without it.
- We expect no CPU or memory problems.

Results

Jetstress HP Sync (100km FC Write Acceleration) passed.

Jetstress HP Sync (100km FC Write Acceleration)

This test runs Microsoft's 2-hour Jetstress Disk Performance Test utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. Jetstress verifies the performance of the disk subsystem as well as the SAN connectivity and replication infrastructure. Jetstress produces an HTML report with such information as overall and per-disk I/Os per second and throughput. Part of the report is very large binary performance log (.blg) files that show performance at a very granular level of detail. These files are not captured as part of this test, but the basic HTML report is. For more information on Jetstress, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27883. Jetstress uses only the database and log devices. In this test, these devices are on a SAN-attached storage frame that is synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is Continous Access XP Sync. FC write acceleration is not enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the Jetstress HP Sync (100km FC Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify FC Write Acceleration is not enabled. Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Run Jetstress on primary Exchange cluster node after verifying through Windows Cluster Administrator that the Cluster and Exchange groups are running there and the database and log devices are on the proper type of replicated storage. Choose 50% for capacity, 100% for throughput, and 2 hours for duration. The test type is performance (rather than streaming backup or soft recovery).

Step 4 After Jetstress is done, check HTML report, switch counters, host status, and replication status and verify expected behavior.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that Jetstress will finish successfully in 2 hours and show less throughput and I/Os per second without FC Write Acceleration enabled than with it.
- We expect no CPU or memory problems.

Results

Jetstress HP Sync (100km FC Write Acceleration) passed.

NetApp

The synchronous replication test for NetApp tests synchronous SnapMirror with and without FC write acceleration.

This section contains the following topics:

Jetstress NetApp Sync (100km FC Write Acceleration)

This test runs Microsoft's 2-hour Jetstress Disk Performance Test utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. Jetstress verifies the performance of the disk subsystem as well as the SAN connectivity and replication infrastructure. Jetstress produces an HTML report with such information as overall and per-disk I/Os per second and throughput. Part of the report is very large binary performance log (.blg) files that show performance at a very granular level of detail. These files are not capatured as part of this test, but the basic HTML report is. For more information on Jetstress, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27883. Jetstress uses only the database and log devices. In this test, these devices are on a SAN-attached storage frame that is synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is NetApp synchronous SnapMirror. FC write acceleration is enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the Jetstress NetApp Sync (100km FC Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify FC Write Acceleration is enabled. Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Run Jetstress on primary Exchange cluster node after verifying through Windows Cluster Administrator that the Cluster and Exchange groups are running there and the database and log devices are on the proper type of replicated storage. Choose 50% for capacity, 100% for throughput, and 2 hours for duration. The test type is performance (rather than streaming backup or soft recovery).

Step 4 After Jetstress is done, check HTML report, switch counters, host status, and replication status and verify expected behavior.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that Jetstress will finish successfully in 2 hours and show about the same throughput and I/Os per second with FC Write Acceleration enabled than without it due to SnapMirror's use of IPFC versus native FC.
- We expect no CPU or memory problems.

Results

Jetstress NetApp Sync (100km FC Write Acceleration) passed.

Jetstress NetApp Sync (100km FC Write Acceleration)

This test runs Microsoft's 2-hour Jetstress Disk Performance Test utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. Jetstress verifies the performance of the disk subsystem as well as the SAN connectivity and replication infrastructure. Jetstress produces an HTML report with such information as overall and per-disk I/Os per second and throughput. Part of the report is very large binary performance log (.blg) files that show performance at a very granular level of detail. These files are not capatured as part of this test, but the basic HTML report is. For more information on Jetstress, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27883. Jetstress uses only the database and log devices. In this test, these devices are on a SAN-attached storage frame that is synchronously replicating to another SAN-attached storage frame in a remote data center separated by 100 km (1 ms round-trip time) of latency. The mechanism used is NetApp synchronous SnapMirror. FC write acceleration is not enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the Jetstress NetApp Sync (100km FC Write Acceleration) test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify FC Write Acceleration is disabled. Periodically gather interface counters, host status, and replication status throughout the test.

Step 3 Run Jetstress on primary Exchange cluster node after verifying through Windows Cluster Administrator that the Cluster and Exchange groups are running there and the database and log devices are on the proper type of replicated storage. Choose 50% for capacity, 100% for throughput, and 2 hours for duration. The test type is performance (rather than streaming backup or soft recovery).

Step 4 After Jetstress is done, check HTML report, switch counters, host status, and replication status and verify expected behavior.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that Jetstress will finish successfully in 2 hours and show about the same throughput and I/Os per second with FC Write Acceleration enabled than without it due to SnapMirror's use of IPFC versus native FC.
- We expect no CPU or memory problems.

Results

Jetstress NetApp Sync (100km FC Write Acceleration) passed.

CHAPTER 16

TIBCO Rendezvous

This phase of testing introduced the TIBCO Rendezvous application into the DCAP data center topology. Hardware consisted of a single client/server pair hosted by HP BL460c, 3.0Ghz Dual-core Intel Xeon, HP blade servers with 8GB of RAM. Baseline tests were first performed between directly connected hosts to qualify the performance with no network transmission delay. After the baselines were completed, testing was performed by running several publisher/subscriber combinations on the client/server pair. The messages were routed through the data center network which was configured with PIM Sparse-mode multicast and Anycast Auto-RP. Latency measurements were calculated by comparing the baseline statistics with the statistics collected when routing through the network. Finally, the data center router's CPU and memory performances were monitored while sending a maximum sustained rate of TIBCO-generated multicast traffic through the network.

TIBCO Rendezvous software makes it easy to deploy distributed applications that exchange data across a network. The Rendezvous daemon runs on each computer involved with message communication. All information that travels between program processes passes through the Rendezvous daemon as the information enters and exits host computers. The daemon also passes information between program processes running on the same host.

TIBCO Rendezvous Concepts

Messages—Carry data among program processes or threads. Messages contain self-describing data fields. Programs can manipulate message fields, send messages, and receive messages.

Events—Create event objects to register interest in significant conditions. For example, dispatching a listener event notifies the program that a message has arrived; dispatching a timer event notifies the program that its interval has elapsed. Programs define event callback functions to process events.

Subjects—Messages are associated with a logical name (subject). Programs listen for a particular subject, or publish messages under a specific subject.

Certified Message Delivery—Confirms delivery of each message to each registered recipient. Certified delivery assures programs that every certified message reaches each intended recipient—in the order sent. When delivery is not possible, both sending and listening programs receive explicit information about each undelivered message.

Programs determine an explicit time limit for each message. After a program sends a certified message, TIBCO Rendezvous software continues delivery attempts until delivery succeeds, or until the message's time limit expires.

TIBCO Rendezvous certified delivery software presents advisory messages to inform programs of every significant event relating to delivery. TIBCO Rendezvous certified delivery software records the status of each message in a ledger. Programs that require certification only for the duration of the program

process should use a process-based ledger. Programs that require certification that transcends process termination and program restart use a file-based ledger. When certified delivery is not allowed, delivery conditions decrease to the standard TIBCO Rendezvous reliable delivery semantics.

Distributed Queue Daemons—Distribute a service over several processes.

The TIBCO Rendezvous daemon completes the information pathway between TIBCO Rendezvous program processes across the network. Programs try to connect to a TIBCO Rendezvous daemon process. If a local daemon process is not yet running, the program starts one automatically and connects to it. The TIBCO Rendezvous daemon arranges the details of data transport, packet ordering, receipt acknowledgement, retransmission requests, and dispatching information to the correct program processes. The daemon hides all these details from TIBCO Rendezvous programs. The TIBCO Rendezvous daemon is almost invisible to the programs that depend on it. Programs send and receive information using TIBCO Rendezvous communications calls, and the TIBCO Rendezvous daemon does the work of getting information to the appropriate place.

The daemon performs the following tasks:

- Transmits outbound messages from program processes to the network.
- Delivers inbound messages from the network to program processes.
- Filters subject-addressed messages.
- Shields programs from operating system idiosyncrasies, such as low-level sockets.

Test Results Summary

Table 16-1 on page 16-3 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 16-1 on page 16-3 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTSs.

Table 16-1 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Latency, page 16-4	n/a	1. Classic RVD Latency DCa to DCa 2. Embedded Daemon Baseline 3. Embedded Daemon Latency DCa to DCa	
Multicast, page 16-9	n/a	Multi Data Center Auto-RP with MSDP Functionality	
Throughput, page 16-12	n/a	1. Maximum Receiving Rate T2A 2. Maximum Sending Rate T1A 3. Maximum Sending Rate T1B 4. Maximum Sustained Rate T3A DCa to DCa 5. Maximum Sustained Rate T3A 6. Maximum Sustained Rate T3B DCa to DCa 7. Maximum Sustained Rate T3B	

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

Latency

Message latency includes the overall latency time from message request from a host client to message receive acknowledgment by the application server. Latency baselines are determined by sending traffic between directly connected client/server hosts. Data Center latency is comprised of the additional latency added when routing the traffic through the Data Center LAN.

This section contains the following topics:

Classic RVD Latency DCa to DCa

Programs using TIBCO depend on the Rendezvous daemon, a background process (rvd), for reliable and efficient network communication. The Rendevous daemon completes the information pathway between Rendezvous program processes across the network. (Usually the daemon runs on the same computer as the program; however, it is possible to connect to a remote daemon.)

The Rendezvous daemon arranges the details of data transport, packet ordering, receipt acknowledgment, retransmission requests, and dispatching information to the correct program processes. The daemon hides all these details from Rendezvous programs.

The Rendezvous daemon is nearly invisible to the programs that depend upon it. Programs send and receive information using Rendezvous communications callls, and the Rendezvous daemon does the work of getting information to the right place.

This test was used to verify the baseline round trip latency statistics for 64 byte multicast messages sent from the TIBCO Rendezvous server and received by the TIBCO Rendezvous client on the classic rvd bus. The TIBCO provided rvlat-classic script was used and run on both clients in both server and client modes. The server listens on TCP and UDP ports 7500 for requests of messages to group 224.1.1.5. The client then requests 100,000 messages of size 64B and the Round Trip Time (RTT) latency for message request, receipt, and acknowledgment is measured.

For this test the servers were placed on different Data Center Vlans so traffic was routed through the LAN. The results were compared to the directly connected client baseline to measure latency added by routing through the Data Center.

Test Procedure

The procedure used to perform the Classic RVD Latency DCa to DCa test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendezvous client and server and verify that they are on the different inband subnets by issuing the **ifconfig** command. This will cause the traffic to pass through the DC LAN and be routed between client and server.

Step 3 Verify in-band connectivity between the client and server with the **ping** command.

Step 4 On both the client and server, change directories to /root/tibco/RVLAT.

Step 5 On the client and server, start the following script:

rvd -reliability 5 &

The script will start the classic rvd listener bus process on both clients with a reliability scope of 5s. This will allow for messages up to 5s to for the TIBCO acknowledgment message to be sent and received.

Step 6 On the server, start the following script:

./rvlat-classic srv -network "eth1;224.1.1.5" -daemon "7500" -service "7500"

The server will begin listening on eth1 for requests to the 224.1.1.5 network on TCP daemon port 7500 and UDP service port 7500. When a data request is made it will respond with the data requested.

Step 7 On the client, start the following script:

./rvlat-classic cli -network "eth1;224.1.1.5" -daemon "7500" -service "7500" -messages "100000" -size "64" > L2BtoA.txt

The client will begin sending requests for 100,000 messages that have a size of 64B to daemon and TCP/UDP service ports 7500. Verify all 100,000 messages are received and that they are within the expected response window.

Copy the results to the Harbor Master directory from the client and server with the scp command and analyze by comparing with the baseline statistics. Quantify the amount of time added to the average message response time by subtracting the baseline from the routed average.

Step 8 Gather the Data Center A(DCA) multicast state by issuing the **show ip pim rp map 224.1.1.5** and **show ip mroute 224.1.1.5** commands on each DCA LAN device.

Step 9 Issue **ctrl+c** on the server side to end the listener script.

Step 10 Issue the same scripts used previously but swap the server/client role. Verify that the stats are similar to what they are in the prior run.

Copy the results to the Harbor Master directory from the client and server with the **scp** command and analyze by comparing with the baseline statistics. Quantify the amount of time added to the average message response time by subtracting the average baseline from the routed average.

Step 11 Gather the DCA multicast state by issuing the **show ip pim rp map 224.1.1.5** and **show ip mroute 224.1.1.5** commands on each DCA LAN device.

Step 12 Kill the rvd process running on each server with the **killall -9 rvd** command.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that no messages will be lost.
- We expect that the average latency per message will be less than 430us.
- We expect both hosts to have similar statistics when acting as the sender/receiver pair.
- We expect that the average amount of time added to the average message response time when routing through the network will be less than 200us.
- We expect no CPU or memory problems.

Results

Classic RVD Latency DCa to DCa passed.

Embedded Daemon Baseline

Programs attempt to connect to a Rendezvous daemon process. If a local daemon process is not yet running, the program starts one automatically and connects to it.

The Rendezvous daemon arranges the details of data transport, packet ordering, receipt acknowledgment, retransmission requests, and dispatching information to the correct program processes. The daemon hides all these details from Rendezvous programs.

The Rendezvous daemon is nearly invisible to the programs that depend upon it. Programs send and receive information using Rendezvous communications callls, and the Rendezvous daemon does the work of getting information to the right place.

This test was used to verify the baseline round trip latency statistics for 64 byte multicast messages sent from the TIBCO server and received by the TIBCO client. The rvlat-embedded script was used and run on both clients in both server and client modes. The rvd process is not manually started which causes the embedded process to be started when the Application script is run. The script on the server listens on TCP and UDP ports 7500 for requests to group 224.1.1.5. The client requests 10000 messages of size 64B and the RTT latency between message request, receipt, and acknowledgment is measured.

For this test the server and client were connected via Layer 2, no routing was done through the DC LAN. This was to baseline the server to client capabilities prior to testing the traffic across the network.

Test Procedure

The procedure used to perform the Embedded Daemon Baseline test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendezvous client and server and verify that they are on the same inband subnet. This will allow for the traffic to pass without routing through the network as the client and server will appear directly connected.

Step 3 Verify in-band connectivity between the client and server with the ping command.

Step 4 On both the client and host, change directories to /root/tibco/RVLAT.

Step 5 On the server, start the following script:

./rvlat-embedded srv -network "eth1;224.1.1.5"

The script will cause the server to send IGMPv2 Membership Reports for group 224.1.1.5 to the embedded rvd daemon which is started when the script is called.

Step 6 On the client, start the following script:

./rvlat-embedded cli -network "eth1;224.1.1.5" -messages "10000" -size "64"

The script will cause the client to send 10000, 64 byte, messages to the 224.1.1.5 group via the embedded rvd daemon which is started when the script is called. The client should recieve 10000 responses back. The statistics form a baseline for client to server latency. With the servers directly connected this time should average to be less than 175us.

Copy the results to the Harbor Master directory from the client and server with the scp command and analyze.

Step 7 Issue ctrl+c on the server side to end the embedded daemon script.

Step 8 Issue the same scripts used previously but swap the server/client role. Verify that the stats are similar to what they were when run previously. Like before, the total request average response time should be less than 175us.

Copy the results to the Harbor Master directory from the client and server with the scp command and analyze.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that no messages will be lost.
- We expect that the average latency per message will be less than 200us.
- We expect both hosts to have similar statistics when acting as the sender.
- We expect no CPU or memory problems.

Results

Embedded Daemon Baseline passed.

Embedded Daemon Latency DCa to DCa

Programs using TIBCO attempt to connect to a Rendezvous daemon process. If a local daemon process is not yet running, the program starts one automatically and connects to it.

The Rendezvous daemon arranges the details of data transport, packet ordering, receipt acknowledgment, retransmission requests, and dispatching information to the correct program processes. The daemon hides all these details from Rendezvous programs.

The Rendezvous daemon is nearly invisible to the programs that depend upon it. Programs send and receive information using Rendezvous communications callls, and the Rendezvous daemon does the work of getting information to the right place.

This test was used to verify the baseline round trip latency statistics for 64 byte multicast messages sent from the TIBCO server and received by the TIBCO client. The TIBCO provided rvlat-embedded script was used and run on both clients in both server and client modes. The rvd process is not manually started which causes the embedded process to be started when the application script is run. The script on the server listens on TCP and UDP ports 7500 for requests to group 224.1.1.5. The client requests 100,000 messages of size 64B and the Round Trip Time (RTT) latency for message request, receipt, and acknowledgment is measured.

For this test the client and server were placed on different Data Center Vlans so traffic was routed through the LAN. The results were compared to the directly connected client baseline to measure latency added by routing through the Data Center.

Test Procedure

The procedure used to perform the Embedded Daemon Latency DCa to DCa test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendezvous client and server and verify that they are on different subnets by issuing the **ifconfig** command. This will allow the traffic to pass through the Data Center LAN and be routed between client and server.

Step 3 Verify in-band connectivity between the client and server with the **ping** command.

Step 4 On both the client and server, change directories to /root/tibco/RVLAT.

Step 5 On the server, start the following script:

./rvlat-embedded srv -network "eth1;224.1.1.5"

The script will cause the server to send IGMPv2 Membership Reports for group 224.1.1.5 to the embedded daemon which is started when the script is called.

Step 6 On the client, start the following script:

./rvlat-embedded cli -network "eth1;224.1.1.5" -messages "10000" -size "64"

The script will cause the client to send 100,000 64 byte messages to the 224.1.1.5 group via the embedded daemon which is started when the script is called. The client should receive 100,000 responses back.

Copy the results to the Harbor Master directory from the client and server with the **scp** command and analyze by comparing with the baseline statistics. Quantify the amount of time added to the average message response time by subtracting the baseline from the routed average.

Step 7 Gather the Data Center A (DCA) multicast state by issuing the **show ip pim rp map 224.1.1.5** and **show ip mroute 224.1.1.5** commands on each DCA LAN device.

Step 8 Issue **ctrl+c** on the server side to end the embedded daemon script.

Step 9 Issue the same scripts used previously but swap the server/client role. Verify that the stats are similar to what they were when run previously.

Copy the results to the Harbor Master directory from the client and server with the **scp** command and analyze by comparing with the baseline statistics. Quantify the amount of time added to the average message response time by subtracting the baseline from the routed average.

Step 10 Gather the DCA multicast state by issuing the **show ip pim rp map 224.1.1.5** and **show ip mroute 224.1.1.5** commands on each DCA LAN device.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that no messages will be lost.
- We expect that the average latency per message will be less than 330us.
- We expect that the average amount of time added to the average message response time when routing through the network will be less than 200us.
- We expect both hosts to have similar statistics when acting as the sender.
- We expect no CPU or memory problems.

Results

Embedded Daemon Latency DCa to DCa passed.

Multicast

PIM Sparse Mode (PIM-SM) uses a pull model to deliver multicast traffic. Only networks that have active receivers that have explicitly requested the data will be forwarded the traffic. PIM-SM is defined in RFC 2362. PIM-SM uses a shared tree to distribute the information about active sources. Depending on the configuration options, the traffic can remain on the shared tree or switch over to an optimized source distribution tree. The latter is the default behavior for PIM-SM on Cisco routers. The traffic starts to flow down the shared tree, and then routers along the path determine whether there is a better path to the source. If a better, more direct path exists, the designated router (the router closest to the receiver) will send a join message toward the source and then reroute the traffic along this path. PIM-SM has the concept of an RP, since it uses shared trees—at least initially. The RP must be administratively configured in the network. Sources register with the RP, and then data is forwarded down the shared tree to the receivers. If the shared tree is not an optimal path between the source and the receiver, the routers dynamically create a source tree and stop traffic from flowing down the shared tree. This is the default behavior in IOS. Network administrators can force traffic to stay on the shared tree by using a configuration option (lp pim spt-threshold infinity). PIM-SM scales well to a network of any size, including those with WAN links. The explicit join mechanism prevents unwanted traffic from flooding the WAN links.

This section contains the following topics:

Multi Data Center Auto-RP with MSDP Functionality

Auto-RP is a feature that automates the distribution of group-to-RP mappings in a PIM network. This feature has the following benefits:

1. It is easy to use multiple RPs within a network to serve different group ranges. 2. It allows load splitting among different RPs and arrangement of RPs according to the location of group participants. 3. It avoids inconsistent, manual RP configurations that can cause connectivity problems.

To make Auto-RP work, a router must be designated as an RP-mapping agent, which receives the RP-announcement messages from the RPs and arbitrates conflicts. The RP-mapping agent then sends the consistent group-to-RP mappings to all other routers. Thus, all routers automatically discover which RP to use for the groups they support.

In the Data Center, each core device is configured to use Auto-RP and Multi Source Discovery Protocol(MSDP). In a multi Data Center deployment a full meshing of RP's with MSDP is configured so that each data center can learn multicast routing information from the neighboring data center.

This test verified the basic functionality of Auto-RP with fully meshed MSDP between two data centers. The configuration on each core router is first shown and the RP mappings on each router in the network is verified. Finally traffic is sent to recievers in each data center and multicast state on each router is verified.

Test Procedure

The procedure used to perform the Multi Data Center Auto-RP with MSDP Functionality test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify the Auto-RP configuration on dca-core-1, dca-core-2, dcb-core-1, and dcb-core-2.

For each Auto-RP candidate, we see the following configuration:

1. A static configuration on loopback 1 defining the anycast address, 10.10.0.1, as the RP

2. All Auto-RP candidates are configured to send RP announce messages (to group 224.0.1.39). The Auto-RP in each data center is announcing its candidacy for all multicast groups.

3. Each RP is also configured as a PIM mapping agent. A mapping agent will listen to group 224.0.1.39 for all candidate RP's. It will then elect an RP based on the highest IP address (in the case of dca-core-1 and dca-core-2, dca-core-2 is selected because its Loopback 0, or Router ID, interface address is higher.) Once it has elected an RP, it will cache RP-to-group mapping information, and send it out periodically to 224.0.1.40, through which all other routers in the network learn the RP-to-group information.

Step 3 Verify that each core device in both DCA and DCB is fully meshed to the other neighboring core device within the same data center and with both core devices in the neighboring data center by issuing the **show ip msdp peer** ***peer ip*** command.

Step 4 Verify each core device shows up as the elected Auto-RP for the data center by issuing the **show ip pim rp mapping** and **show ip pim rp** commands.

In Data Center's A and B, dca-core-2 and dcb-core-2 are the active RP's, respectively.

Step 5 Verify the RP information is passed downstream to the aggregation switches by issuing the **show ip pim rp mapping** and **show ip pim rp** commands.

Step 6 Begin sending traffic with the Shenick test tool. Traffic will be source from a single host in each data center. The DCa source is advertising 10 muticast groups(239.100.1.[1-10]) and the DCb source is also advertising 10 multicast groups(239.200.1.[1-10]). The stream also consists of a multicast receiver for these groups.

Verify the traffic is received correctly by the receiver in each Data Center on the Shenick tool by verifying the streams become colored green.

Step 7 Verify that the DCA and DCB core devices have correct mroute entries and flags for the test traffic groups and that traffic is being hardware switched.

For groups 239.100.0.[1-10], dca-core-2 is the PIM-RP. Being the PIM-RP, we expect dca-core-2 to have an (S,G) entry for each of these 10 groups. We expect to see the following flags on each entry: a T-flag indicating that the Shortest-Path Tree (SPT) is being used. The incoming interface for all 10 groups should be TenGigabitEthernet1/3. GigabitEthernet5/1 should be in the Outgoing Interface List (OIL). An RPF-MFD tag should be on each mroute entry indicating that the entry is hardware-switching capable. An H-flag should accompany each interface in the OIL, indicating that traffic out that interface is being hardware-switched. The MMLS entries should be consistent with the mroute entries.

For groups 239.200.0.[1-10], dcb-core-2 is the PIM-RP. Being the PIM-RP, we expect dcb-core-2 to have an (S,G) entry for each of these 10 groups. We expect to see the following flags on each entry: a T-flag indicating that the Shortest-Path Tree (SPT) is being used. The incoming interface for all 10 groups should be TenGigabitEthernet1/3. GigabitEthernet5/1 should be in the Outgoing Interface List (OIL). An RPF-MFD tag should be on each mroute entry indicating that the entry is hardware-switching capable. An H-flag should accompany each interface in the OIL, indicating that traffic out that interface is being hardware-switched. The MMLS entries should be consistent with the mroute entries.

Step 8 Verify that the DCA and DCB aggregation devices have correct mroute entries and flags for the test traffic groups and that traffic is being hardware switched.

dca-agg-2 is first-hop router for traffic groups 239.100.0.[1-10]. As such, it should be registered with the RP, 10.10.0.1. The first-hop router, and all routers up to and including the RP, will have an (S,G) entry for a given group, as per the rules of PIM sparse-mode. dca-agg-2 should have the (S,G) entry for each group in the test range. It should also have the F and T flags set for this (S,G) entry. The F flag is set because the source is registered with the RP. The T flag is set because a Shortest-Path Tree (SPT) is formed from the source to the RP. The Incoming interface is VLAN1133. The outgoing interface is TenGigabitEthernet 9/2, as this is the best path to dca-core-2, the primary PIM-RP. The outgoing interface should also have the H flag set, indicating that it is being hardware-switched. Each group should have an MMLS entry with the (S,G) entry. This (S,G) in the MMLS entry should have incoming and outgoing interfaces listed that are consistent with the mroute (S,G) entry.

dcb-agg-1 is first-hop router for traffic groups 239.200.0.[1-10]. As such, it should be registered with the RP, 172.30.0.201. The first-hop router, and all routers up to and including the RP, will have an (S,G) entry for a given group, as per the rules of PIM sparse-mode. dca-agg-1 should have the (S,G) entry for each group in the test range. It should also have the F and T flags set for this (S,G) entry. The F flag is set because the source is registered with the RP. The T flag is set because a Shortest-Path Tree (SPT) is formed from the source to the RP. The Incoming interface is VLAN1133. The outgoing interface is TenGigabitEthernet3/3, as this is the best path to dcb-core-2, the primary PIM-RP. The outgoing interface should also have the H flag set, indicating that it is being hardware-switched. Each group should have an MMLS entry with the (S,G) entry. This (S,G) in the MMLS entry should have incoming and outgoing interfaces listed that are consistent with the mroute (S,G) entry.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that each data center will be configured with Auto-RP.
- We expect that the RP's in each data center will be fully meshed with MSDP.
- We expect that the basic functionality of the dual data center PIM deployment will pass test traffic as expected.
- We expect that the mroute state will be correct on all routers in each data center.

- We expect no CPU or memory problems.

Results

Multi Data Center Auto-RP with MSDP Functionality passed.

Throughput

Throughput is a measure of the maximum sending and receiving rate of the TIBCO client and server. Throughput baselines are determined by sending traffic between directly connected client/server hosts. Throughput effects on the network were measured by monitoring CPU/Memory usage on the Data Center LAN devices while sending traffic between the client/server hosts at the maximum sustainable rate.

This section contains the following topics:

Maximum Receiving Rate T2A

This test was used to verify maximum receiving rate a system was capable of receiving messages at. For this test system A ran 1 RVD daemon and 1 Application service to send traffic to 8 subscribers on system B.

Test Procedure

The procedure used to perform the Maximum Receiving Rate T2A test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendevous server(System A) and client(System B) and verify eth1 is configured on the inband network.

Step 3 On both systems, change directories to /root/tibco/THRUPUT.

Step 4 On both systems start the following script:

rvd -reliability 5 -listen 7501 &

The script will start the class rvd listener bus process listening on TCP port 7501 on the system with a reliability of 5s.

Step 5 On System B start the following script:

./rvperf-sustained-sub_t2a.bash

The script will start 8 copies of rvperfs which are message subscribers. Use the command **ps -ef | grep rv** to verify 8 instances of this have started. If there are any issues issue the **kill -9 rvperfs** command and restart the script.

Step 6 On System A start the following script:

./rvperf-sustained_t2a.bash

This will start sending messages of sizes 64B, 500B, 1350B, 2KB, 10KB, 50KB, 1MB, and 1.2MB to the listening receivers.

Step 7 Upon completion issue the following command on system B to gather the results:

grep Messages/second rvperfsrun*.txt >> step7-t2a_results.txt

Copy the results to the correct Harbor Master directory and anylyze.

Step 8 On system A copy the following files to the results directory:

t2a-results.pub.txt t2b-results.sub.txt

If t2a_results.sub.txt if missing some data points retrieve from the t2a_results.txt file.

Step 9 Kill all rvperfs and rvd running processes on the client by issuing the **killall -9 rvperfs** command. Verify all processes have been killed with the **ps -ef | grep rv** command.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the client will send messages of 64B, 500B, 1350B, 2KB, 10KB, 50KB, and 100KB.
- We expect the script to complete without maxing out the systems CPU utilization.
- We expect no CPU or memory problems.

Results

Maximum Receiving Rate T2A passed.

Maximum Sending Rate T1A

This test was used to verify the maximum sending rate a system was capable of sending messages at. For this test, System A, the "server," runs 1 rvd daemon and 8 Application services. This traffic is destined for 1 Publisher/Service. Since the test is only concerned with the sending rate of the system sending messages the listener system, system B, is virtual.

Test Procedure

The procedure used to perform the Maximum Sending Rate T1A test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendevous server(System A) and verify eth1 is configured on the inband network.

Step 3 On System A, change directories to /root/tibco/THRUPUT.

Step 4 On System A start the following script:

rvd -reliability 5 -listen 7501 &

The script will start the class rvd listener bus process listening on TCP port 7501 on the system with a reliability of 5s.

Step 5 On System A start the following script:

./rvperf_t1a.bash

The script will start 8 application daemon's sending messages of increasing sizes: 64B, 500B, 1350B, 2KB, 10KB, 50KB, and 100KB. All messages of one size will be sent, then the message size will increase.

Step 6 Upon completion copy the results t1a_results.txt from the client to the results directory. The data gathered is a baseline for the maximum send rate the server is capable of for each message size.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the client will send messages of 64B, 500B, 1350B, 2KB, 10KB, 50KB, and 100KB.
- We expect to baseline the servers maximum sending rate.
- We expect no CPU or memory problems.

Results

Maximum Sending Rate T1A passed.

Maximum Sending Rate T1B

In most situations, each host runs one Rendezvous daemon process, however, multiple rvd daemon's can be run on a host.

This test was used to verify maximum sending rate a system was capable of sending messages at using multiple rvd daemons. For this test system A used 4 rvd daemon's and 8 Application services to send traffic to 1 Publisher/Service. Since the test is only concerned with the sending rate of the system sending messages the listener system, system B, is virtual.

Test Procedure

The procedure used to perform the Maximum Sending Rate T1B test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendevous server(System A) and verify eth1 is configured on the inband network.

Step 3 On System A, change directories to /root/tibco/THRUPUT.

Step 4 On System A start the following script:

rvd -reliability 5 -listen 7501 & rvd -reliability 5 -listen 7502 & rvd -reliability 5 -listen 7503 & rvd -reliability 5 -listen 7504 &

The script will start the class rvd listener bus process listening on TCP port 7501-7504 on the system with a reliability of 5s.

Step 5 On System A start the following script:

./rvperf_t1b.bash

The script will start 8 application daemon's sending messages of increasing sizes: 64B, 500B, 1350B, 2KB, 10KB, 50KB, and 100KB. All messages of one size will be sent, then the message size will increase.

Step 6 Upon completion copy the results t1b_results.txt from the client to the results directory.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the client will send messages of 64B, 500B, 1350B, 2KB, 10KB, 50KB, and 100KB.
- We expect to baseline the servers maximum sending rate when using multiple rvd daemons.
- We expect no CPU or memory problems.

Results

Maximum Sending Rate T1B passed.

Maximum Sustained Rate T3A DCa to DCa

This test was used to verify maximum sustaining rate the systems were capable of sending/receiving messages at when located in the same Data Center. For this test the Server, System A, started 1 RVD daemon, 8 Application services, and 1 Publisher/Service to respond to message requests. The Client, System B, started 1 RVD daemon, 8 Application Service, and 1 Publisher/Service to request messages of 64B, 500B, 1350B, 2KB, 50MB, and 100MB.

This test was designed to stress and verify the network did not suffer unacceptable memory or CPU impact when sending the maximum sustained rate traffic through the DCA network from client to server.

Test Procedure

The procedure used to perform the Maximum Sustained Rate T3A DCa to DCa test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendezvous server and client and verify that they are on located in the same Data Center but on different subnets by issuing the **ifconfig** command. This will cause traffic between hosts to be routed through the DC LAN.

Step 3 On both systems, change directories to /root/tibco/THRUPUT.

Step 4 On both the client and server start the following script:

rvd -reliability 5 -listen 7501 &

The script will start the class rvd listener bus process listening on TCP port 7501 on the system with a reliability of 5s.

Step 5 On the client start the following script:

./rvperf-sustained-sub_t3a.bash

The script will start 8 copies of rvperfs which are message subscribers that will request messages of sizes 64B, 500B, 1350B, 2KB, 10KB, 50KB, 100KB from the 8 application services sending to 239.1.1.1-239.1.1.8. Use the command **ps -ef | grep rv** to verify 8 instances of this have started. If there are any issues issue the **kill -9 rvperfs** command and restart the script.

Step 6 On the server start the following script:

./rvperf-sustained_t3a.bash

This will source messages of sizes 64B, 500B, 1350B, 2KB, 10KB, 50KB, 100KB to the 8 application services groups 239.1.1.1-239.1.1.8.

Step 7 While the traffic is running, verify the multicast state state on each device in both Data Centers by issuing the **show ip pim rp map** and **show ip mroute** commands. The test runs for approximately 1 hour. Check the mcast state every 30 minutes to verify state remains consistent.

Step 8 Upon completion issue the following command on the client to gather the results:

grep Messages/second rvperfsrun*.txt >> step8-t3a_results.txt

Copy the results to the correct Harbor Master directory.

Step 9 On the server copy the following files to the results directory:

t3a-results.pub.txt t3a-results.sub.txt

If t3a_results.sub.txt if missing some data points retrieve from the t3a_results.txt file. The files will report the amount of data traffic sent through the network by the scripts.

Step 10 Kill all rvperfs running processes on the client by issuing the **killall -9 rvperfs** command. Verify all runinng processes have been killed with the **ps -ef | grep rv** command.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect 8 application subscribers on System B to request messages of 64B, 500B, 1350B, 2KB, 50MB, and 100MB.
- We expect that the 8 application clients on System A will respond to message requests from System B.
- We expect no CPU or memory problems.

Results

Maximum Sustained Rate T3A DCa to DCa passed.

Maximum Sustained Rate T3A

This test was used to verify maximum sustained the systems were capable of sending/receiving messages at. For this test the Server, System A, ran 1 RVD daemon, 8 Application services, and 1 Publisher/Service to respond to message requests. The Client, System B, ran 1 RVD daemon, 8 Application Service, and 1 Publisher/Service to request messages of 64B, 500B, 1350B, 2KB, 50MB, and 100MB. The maximum sustained rate was baselined by using a directly connected server and client.

Test Procedure

The procedure used to perform the Maximum Sustained Rate T3A test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendevous server(System A) and client(System B) and verify eth1 is configured on the inband network.

Step 3 On both systems, change directories to /root/tibco/THRUPUT.

Step 4 On both systems start the following script:

rvd -reliability 5 -listen 7501 &

The script will start the class rvd listener bus process listening on TCP port 7501 on the system with a reliability of 5s.

Step 5 On System B start the following script:

./rvperf-sustained-sub_t3a.bash

The script will start 8 copies of rvperfs which are message subscribers that will request messages of sizes 64B, 500B, 1350B, 2KB, 10KB, 50KB, 100KB from the 8 application services sending to 239.1.1.1-239.1.1.8. Use the command **ps -ef | grep rv** to verify 8 instances of this have started. If there are any issues issue the **kill -9 rvperfs** command and restart the script.

Step 6 On System A start the following script:

./rvperf-sustained_t3a.bash

This will source messages of sizes 64B, 500B, 1350B, 2KB, 10KB, 50KB, 100KB to the 8 application services groups 239.1.1.1-239.1.1.8.

Step 7 Upon completion issue the following command on system B to gather the results:

grep Messages/second rvperfsrun*.txt >> step7-t3a_results.txt

Copy the results to the correct Harbor Master directory.

Step 8 On system A copy the following files to the results directory:

t3a-results.pub.txt t3a-results.sub.txt

If t3a_results.sub.txt if missing some data points retrieve from the t3a_results.txt file.

Step 9 Kill all rvperfs running processes on the client by issuing the **killall -9 rvperfs** command. Verify all runinng processes have been killed with the **ps -ef | grep rv** command.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect 8 application subscribers on System B to request messages of 64B, 500B, 1350B, 2KB, 50MB, and 100MB.
- We expect that the 8 application clients on System A will respond to message requests from System B.
- We expect no CPU or memory problems.

Results

Maximum Sustained Rate T3A passed.

Maximum Sustained Rate T3B DCa to DCa

This test was used to verify maximum sustained the systems were capable of sending/receiving messages at. For this test System A used 4 RVD Daemons, 2 Services/Daemon, and 8 Publishers to send messages to multicast groups 239.1.1.1-239.1.1.8 on ports 7501-7504. System B used 4 RVD Daemons, 2 Services/Daemon, and 8 subscribers to make data requests from groups 239.1.1.1-239.1.1.8 on ports 7501-7504. The servers were placed on different DC Vlan's so traffic was routed through the DCA LAN.

Test Procedure

The procedure used to perform the Maximum Sustained Rate T3B DCa to DCa test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendezvous server(System A) and client(System B) and verify that they are on located in the same Data Center but on different subnets. This will cause traffic between hosts to be routed through the DC LAN.

Step 3 On both systems, change directories to /root/tibco/THRUPUT.

Step 4 On both systems start the following scripts:

rvd -reliability 5 -listen 7501 & rvd -reliability 5 -listen 7502 & rvd -reliability 5 -listen 7503 & rvd -reliability 5 -listen 7504 &

The script will start the classic rvd listener bus process listening on TCP ports 7501-7504 on the system with a reliability of 5s.

Step 5 On System B start the following script:

./rvperf-sustained-sub_t3b.bash

The script will start 8 copies of rvperfs which are message subscribers that will request messages of sizes 64B, 500B, 1350B, 2KB, 10KB, 50KB, 100KB from the 8 application services sending to 239.1.1.1-239.1.1.8. Use the command **ps -ef | grep rv** to verify 8 instances of this have started. If there are any issues issue the **kill -9 rvperfs** command and restart the script.

Step 6 On System A start the following script:

./rvperf-sustained_t3b.bash

This will source messages of sizes 64B, 500B, 1350B, 2KB, 10KB, 50KB, 100KB to the 8 application services groups 239.1.1.1-239.1.1.8.

Step 7 While the traffic is running, verify the multicast state state on each device in both Data Centers by issuing the **show ip pim rp ma** and **show ip mroute** commands. The test runs for approximately 1 hour. Check the mcast state every at the beginning and again 30 minutes into the test to verify state remains consistent.

Step 8 Upon completion issue the following command on system B to gather the results:

grep Messages/second rvperfsrun*.txt >> step8-t3b_results-log.txt

Copy the results to the correct Harbor Master directory.

Step 9 On system A copy the following files to the results directory:

t3b-results.pub.txt t3b-results.sub.txt

If t3b_results.sub.txt if missing some data points retrieve from the t3b_results.txt file.

Step 10 Kill all rvperfs running processes on the client by issuing the **killall -9 rvperfs** command. Verify all runinng processes have been killed with the **ps -ef | grep rv** command.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the 8 application clients on System A will send messages to the 4 local RVD busses.
- We expect the 4 remote RVD busses to collect the sent messages and distribute among the 8 subscribers.
- We expect no CPU or memory problems.

Results

Maximum Sustained Rate T3B DCa to DCa passed.

Maximum Sustained Rate T3B

This test was used to verify maximum sustained the systems were capable of sending/receiving messages at. For this test System A used 4 RVD Daemons, 2 Services/Daemon, and 8 Publishers to send messages to multicast groups 239.1.1.1-239.1.1.8 on ports 7501-7504. System B used 4 RVD Daemons, 2 Services/Daemon, and 8 subscribers to make data requests from groups 239.1.1.1-239.1.1.8 on ports 7501-7504.

Test Procedure

The procedure used to perform the Maximum Sustained Rate T3B test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 SSH to the TIBCO Rendevous server(System A) and client(System B) and verify eth1 is configured on the inband network.

Step 3 On both systems, change directories to /root/tibco/THRUPUT.

Step 4 On both systems start the following scripts:

rvd -reliability 5 -listen 7501 & rvd -reliability 5 -listen 7502 & rvd -reliability 5 -listen 7503 & rvd -reliability 5 -listen 7504 &

The script will start the classic rvd listener bus process listening on TCP ports 7501-7504 on the system with a reliability of 5s.

Step 5 On System B start the following script:

./rvperf-sustained-sub_t3b.bash

The script will start 8 copies of rvperfs which are message subscribers. Use the command **ps -ef | grep rv** to verify 8 instances of this have started. If there are any issues issue the **kill -9 rvperfs** command and restart the script.

Step 6 On System A start the following script:

./rvperf-sustained_t3b.bash

This will start sending messages of sizes 8 application services to the 8 listening recievers. There are two applications services sending to each rvd bus.

Step 7 Upon completion issue the following command on system B to gather the results:

grep Messages/second rvperfsrun*.txt >> step7-t3b_results.txt

Copy the results to the correct Harbor Master directory.

Step 8 On system A copy the following files to the results directory:

t3b-results.pub.txt t3b-results.sub.txt

If t3b_results.sub.txt if missing some data points retrieve from the t3b_results.txt file.

Step 9 Kill all rvperfs running processes on the client by issuing the **killall -9 rvperfs** command. Verify all runinng processes have been killed with the **ps -ef | grep rv** command.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that the 8 application clients on System A will send messages to the 4 local RVD busses.
- We expect the 4 remote RVD busses to collect the sent messages and distribute among the 8 subscribers.
- We expect message loss to be acceptable.
- We expect no CPU or memory problems.

Results

Maximum Sustained Rate T3B passed.

CHAPTER 17

Disaster Recovery—High Availability

DCAP Phase 4 testing included disaster recovery and high availability testing for the Oracle 11i E-Business Suite, Oracle 10gR2 database, and Microsoft Exchange 2003 applications as well as standalone replication testing for NetApp SnapMirror over IP with WAAS optimization using both ACE and WCCPv2 redirection.

The application topology consists of a dual data center Oracle E-Business Suite and a Microsoft Exchange environment. The following two sections detail the application-specific components of the test topology that are important for disaster recovery testing. For more detailed information about the LAN, CSM, GSS, WAAS, SAN, Oracle, or Exchange components, please see the appropriate section in the DCAP documentation for this phase.

Oracle E-Business Suite Environment

The Oracle E-Business environment consists of three tiers: a Database Tier, an Application Tier, and a Desktop Tier.

The database tier consists of an active/active 10gR2 Real Application Cluster (RAC) cluster in Data Center A (DCa) and an active/passive RedHat Cluster Suite (RHCS) cluster in DCb. All hosts are running the RedHat Enterprise Linux 4 update 4 operating system. The hosts in each cluster access the Oracle executable code from a NetApp NAS filer in each data center (each cluster has a separate filer). The database itself, called OEFIN, is normally located on SAN storage in DCa on storage arrays from EMC, HP, and NetApp. This storage uses each vendor's synchronous replication method (SRDF/S for EMC, Continuous Access XP Synchronous for HP, and synchronous SnapMirror for NetApp over an extended fibre channel link to ensure an identical copy of the database is available for failover in DCb.

The Application Tier consists of four non-clustered application hosts in each data center. Three application hosts provide web and Oracle Forms access and the fourth application host runs the Oracle Concurrent Manager (batch processor). All eight hosts share a single network attached storage (NAS) volume on a NetApp filer cluster. This volume, commonly called "APPL_TOP" (pronounced "apple top") after the default top-level directory name, contains Oracle executable code as well as log and output directories. The reasons for using the same NAS volume are three-fold: (1) to allow log and output files written by the Concurrent Manager hosts to be available from any of the application hosts; (2) to allow additional application nodes to be added easily to the Application Tier; and (3) to simplify code management (software patches and upgrades). Normally this volume is available only in DCa. To enable failover to DCb, the volume is synchronously replicated over an IP WAN link using synchronous SnapMirror to a NetApp filer cluster in DCb.

The Desktop Tier consists of a Microsoft Windows host in each of the three branches with a web browser (Internet Explorer version 6). Oracle clients use the web browser with the HTTP protocol to access the applications URL at http://wwwin-oefin.gslb.dcap.com. All client traffic is optimized by WAAS. All clients are located on the DCAP intranet; no Internet clients have access, and therefore no advanced services like firewalls or proxies are needed.

A pair of Application Control Engine (ACE) modules in DCa and a pair of Content Switching Modules (CSMs) in DCb provide load balancing among the three application hosts in each data center by means of separate virtual IPs (VIPs), one for DCa and the other for DCb. The VIPs for each data center as well as the Oracle database and the NAS volume for the shared APPL_TOP directory are in different networks since there is no L2 adjacency between the data centers and no advanced capabilities like route health injection are being used.

A pair of Global Site Selectors (GSS) in each data center together provide automatic load balancing and failover of the ACE and CSM VIPs for the application hosts to support the Oracle clients at all three branch locations. The GSSs monitor not only the Oracle application VIP, but also the Oracle database IP address and the NetApp NAS IP address for the APPL_TOP volume to support failover logic. The reason for monitoring all three IP addresses at each data center even though only the application VIPs are active in both data centers is to enable GSS to failover all the addresses dynamically. The actual failover decision is based only on the health of the ACE and CSM VIPs. In other words, if the ACE VIP goes down in DCa, the remaining GSSes in DCb assume DCa is down entirely. The design also supports an orderly failback to DCa.

Microsoft Exchange Environment

The Microsoft Exchange environment consists of the following key components:

- Four Microsoft Windows 2003 active/passive back end clusters, two in each data center, which host the Microsoft Exchange 2003 application; one cluster in each data center is on physical rack mounted hosts and the other cluster is on VMware virtual hosts on HP c-Class BladeSystem hosts.
- An Outlook 2003 client in each of the three branches.
- A Linux server in each branch which sends SMTP email to all three clients throughout each test.

The primary physical cluster in DCa hosts an Exchange Virtual Server called "DCAP-MBOX-1" and the other physical cluster in DCb acts as a disaster recovery standby cluster. The primary virtual cluster in DCb hosts an Exchange Virtual Server called "DCAP-MBOX-2" and the other virtual cluster in DCa acts as a disaster recovery standby cluster. The clusters use fibre channel to attach to storage from EMC, HP, and NetApp. As in the Oracle environment, this storage is replicated synchronously from the primary to the standby cluster. The Exchange clients use Microsoft Outlook 2003 and the MAPI protocol to access the applications. The Linux hosts use a simple Perl script to send email repeatedly to all clients. All client traffic is optimized by WAAS. All DNS services for the Exchange Virtual Servers are provided by GSS. All clients are located on the DCAP intranet; no Internet clients have access, and therefore no advanced services like firewalls or proxies are needed. The current design supports a manual failover and failback of all components.

Disaster Recovery Testing

The data center disaster recovery tests include failing both applications over to DCb, and then failing the applications back to DCa. Failover testing starts by simulating a disaster by severing all WAN and SAN links to and from DCa. Failback testing starts with a controlled shutdown of applications in DCb. Application data created or modified in DCb during failover is replicated back to DCa as part of the failback procedure.

Both the failover and failback procedures are partly automatic and partly manual. The primary automatic components include the following:

- For Oracle only, recognition by GSS that the CSM VIP becomes unavailable at the start of a failover or failback, referring Oracle clients to a "sorry server" until the application is available, and referring Oracle and NAS clients to the correct IP addresses once the application is available.
- Recognition by the storage arrays that replication has faulted and putting the failover replica storage in a state that allows an administrator to make it available to the failover application clusters.

The key manual components include the following:

- Issuing the storage vendor-specific commands to make the appropriate storage available for the application clusters.
- Starting up the applications.
- For Microsoft Exchange only, pushing the IP address change information for the Exchange Virtual Server to the branch DNS servers.

A little more detail about the Oracle GSS functionality is important to understanding how the automated and manual components work together. All four GSSs were authoritative for the domain gslb.dcap.com, which includes the application host name "wwwin-oefin.gslb.dcap.com" as well as the Oracle database listener host name "db-oefin.gslb.dcap.com" and the NAS host name "nas-oefin.gslb.dcap.com." A client DNS request to resolve the application host name to an IP address arrives at each of the four GSSs by means of name server forwarding configured on the client’s local branch name server. Once the DNS request arrives at one of the four GSSs, the GSS hands out either the VIP for DCa or DCb depending upon the health of the VIPs in each data center.

During normal operations, the GSS's give application clients the appropriate VIP for the wwwin-oefin host name based on the load balancing algorithm chosen by the administrator. These tests used the Weighted Round Robin load balancing type to direct 80% of the clients to DCa and 20% to DCb. The reason the load is lopsided is to keep the majority of the clients in the data center where all the services components are active but send enough clients through the other data center to ensure the failover configuration is working properly. The GSS's always give the DCa NAS and database listener IP addresses for the nas-oefin and db-oefin host names, respectively.

During failover, when the GSS detects the VIP in DCa is down, it initiates failover of all the hostnames associated with the application services. Note that during failover, the DCa GSS's are not available and do not participate in DNS resolution. Because failing over the storage (both SAN and NAS) is manual in the current design, the DCb GSS's redirect client requests to a "sorry server" that displays a web page to users with a standard "service not available" message. The GSS's immediately gives the DCb NAS and database IP addresses for the nas-oefin and db-oefin host names, respectively, even though technically this isn't required until after the NAS and SAN storage is failed over. After the NAS storage is failed over, APPL_TOP must be remounted on the application servers. (NOTE: The remount might fail and a reboot of an application host might be required if it hangs due to performing an I/O operation to APPL_TOP at the exact moment DCa became unavailable; this is an operating system/NFS limitation.) After the SAN storage is failed over, the database file systems must be mounted and the database and listener started up. To bring up the Concurrent Manager host, because Parallel Concurrent Processing is not enabled in the current design, a manual step to update the FND_NODES table is

required to register the DCb Concurrent Manager host. Once all the necessary application services (Oracle database and listener as well as at least one web and Oracle Forms server and the Concurrent Manager server) are up, the CSM brings up the VIP in DCb. When the GSS's detect that the VIP is up, they direct all client requests to the DCb CSM VIP.

Sometime before failback, the Oracle application components are verified as being down in DCa before connectivity to DCa is restored for both IP and SAN extension. This ensures all four GSS's give out consistent responses for all DNS queries. For the HP storage test, replication of the DCb data is then started (since updating DCa storage with just the changes made in DCb during failover is not supported). In this case, failback is not started until the DCb to DCa replication is in fully synchronous mode (that is, the initial copy is complete). This step is not required for EMC and NetApp storage, since the assumption is made that the storage devices in DCa are intact after the simulated disaster is over and the replication mechanisms are aware of what data changed during failover. If this assumption is not made, then this step is required for all storage vendors.

During failback, the Oracle application components are gracefully brought down in DCb. As soon as the DCb CSM takes the Oracle application VIP offline, all GSS's refer Oracle clients to the "sorry server." As soon as storage is failed back to DCa and the database and listener are online, the application is brought manually on all Oracle application servers. The CSMs in each data center bring the Oracle VIP online as soon as at least one web and Oracle Forms server and the Concurrent Manager server are up. When the first CSM is up, all GSS's refer clients to that VIP. Once both CSM VIPs are up, the GSS's return to normal operation and load balance across both data centers.

For additional details on how GSS is configured for Oracle failover and failback, please refer to the GSS and Oracle sections of this document.

There are two key metrics which determine the success or failure of the failover and failback:

- Recovery Point Objective (RPO): this is a measure of the amount of transactions lost in terms of the difference in the currency of the data between the primary data and the failover replica.
- Recovery Time Objective (RTO): this is a measure of the amount of time application data is unavailable to clients.

To allow determination of RPO and RTO, some key data points are gathered throughout the test.

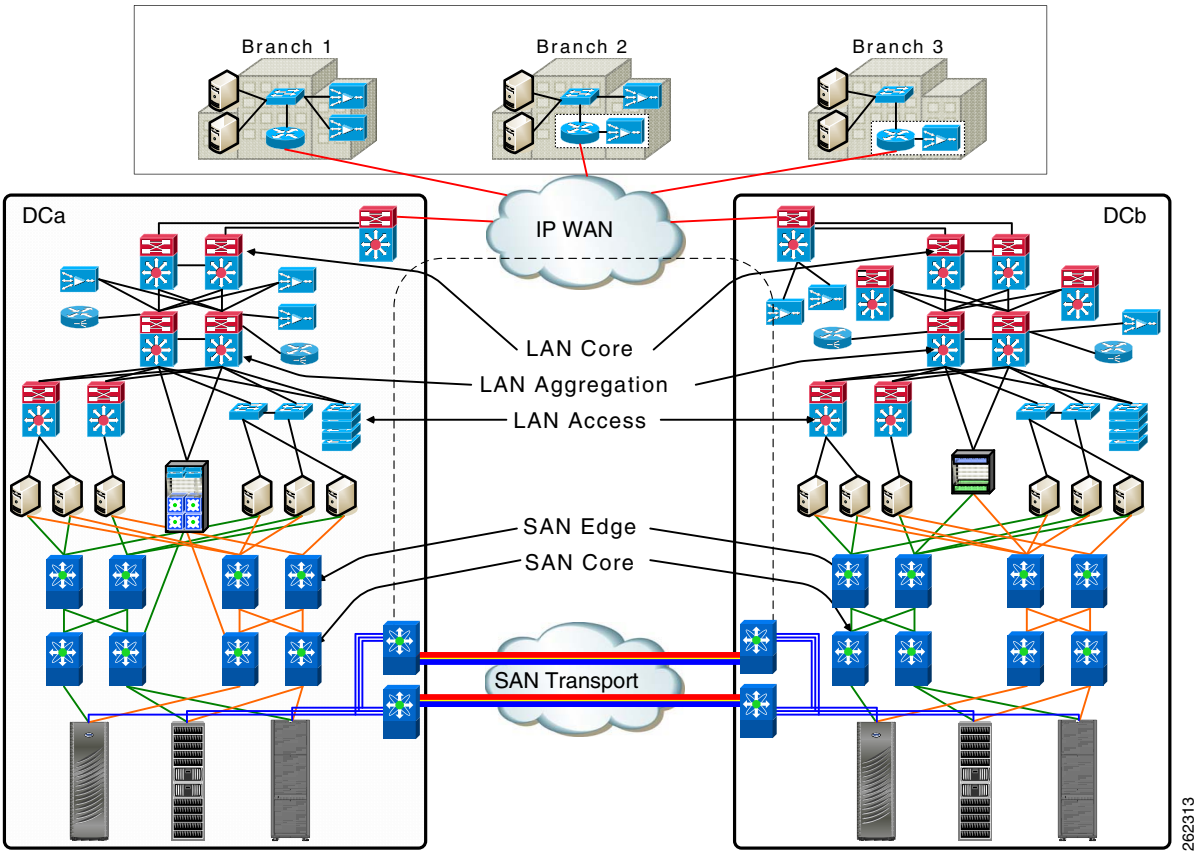
For RPO, at the start of each test, application data is generated by simulated clients (LoadRunner for Oracle and a custom Perl script for Exchange), and a check is made at the completion of failover or failback to ensure all data sent prior to the beginning of the failover or failback is available.

For RTO, the time is measured from the beginning of a failover (when the simulated disaster begins) or failback (when the planned downtime window starts) to the end (when the last application is available to at least one client). The times when all services are up and all clients have access are also tracked. Intermediate times for all key milestones are also tracked to help analyze any unforeseen delays that may have occurred.

Data Center Disaster Recovery Topology

Figure 17-1 depicts the primary data center components that are important in understanding the DCAP data center disaster recovery failover and failback testing.

Figure 17-1 DCAP Data Center Disaster Recovery Topology



The components are detailed in other sections of this document. In sum, they include the following:

- A primary active/passive Oracle database cluster in DCa and a similarly configured failover active/passive cluster in DCb.
- Two Oracle web application hosts and one Oracle concurrent manager hosts in both data centers.
- A NetApp NAS filer cluster in each data center to provide shared Oracle code file systems for the database and application hosts and a shared Oracle data file system ("APPL_TOP") for the Oracle web application hosts.
- CSM and GSS devices in both data centers to provide load balancing and failover.
- WAAS devices in both data centers to provide application acceleration using TFO, DRE, compression, and WAFS (for NFSv2 over TCP access by the Oracle web application host to the NetApp NAS filer).
- One master DNS server and one Windows Domain Controller server per data center providing name to IP address resolution and Active Directory (AD) services. (The Windows servers also act as secondary DNS servers and WINS servers.)

- One secondary DNS Windows server per branch, which are automatically updated through zone file transfers. Branch client hosts use the local secondary DNS server for queries.
- SAN storage connected and replicated through the DCAP SAN testbed in both data centers. The Exchange data is located on SAN storage that's synchronously replicated over a simulated 100 km distance from DCa to DCb. Fiber channel-attached storage from three different vendors, EMC, Hewlett Packard, and NetApp, is used, and the replication mechanism is SRDF/S, Continuous Access XP Synchronous, and synchronous SnapMirror, respectively.
- Oracle web clients and Microsoft Outlook 2003 clients in each branch. The Outlook clients access the back-end Exchange server using the MAPI protocol over the DCAP testbed WAN infrastructure, which incorporates WAAS to optimize MAPI traffic.

For failover, three separate tests, one for each storage vendor, are performed. The sole differences among the tests are the type of SAN storage used and method for failing over SAN storage. The tests consist of application clients initiating transactions continuously from each of the three branch locations. After a short time, a failure of DCa is simulated by severing all its WAN and SAN replication links. After the failover for each application is done, the data is checked to make sure data from all transactions completed prior to the point of the failure is available in DCb. This determines the RPO. The RTO is determined by measuring the time from the failure to the time when the last application becomes available to at least one client in DCb.

For failback, three separate tests, one for each storage vendor, are performed. The sole differences among the tests are the type of SAN storage used and method for failing back SAN storage. The tests essentially reverse the effect of the fail over test, except rather than a simulated failure being the first step, a controlled shutdown of applications starts the test. After the failback for each application is done, the data is checked to make sure data from all transactions completed prior to the start of the shutdown is available in DCa. This determines the RPO. The RTO is determined by measuring the time from the shutdown to the time when the last application becomes available to at least one client in DCa.

Table 17-1 summarizes the results of failover and failback testing.

Table 17-1 Failover/Failback Test Results

	Failover		Failback	
Vendor	**RPO**	**RTO**	**RPO**	**RTO**
EMC	0	14 min	0	21 min
HP	0	13 min	0	19 min
NetApp	0	14 min	0	19 min

Figure 17-2 through Figure 17-7 show separate timeline details for each test.

Figure 17-2 Failover Timeline—EMC

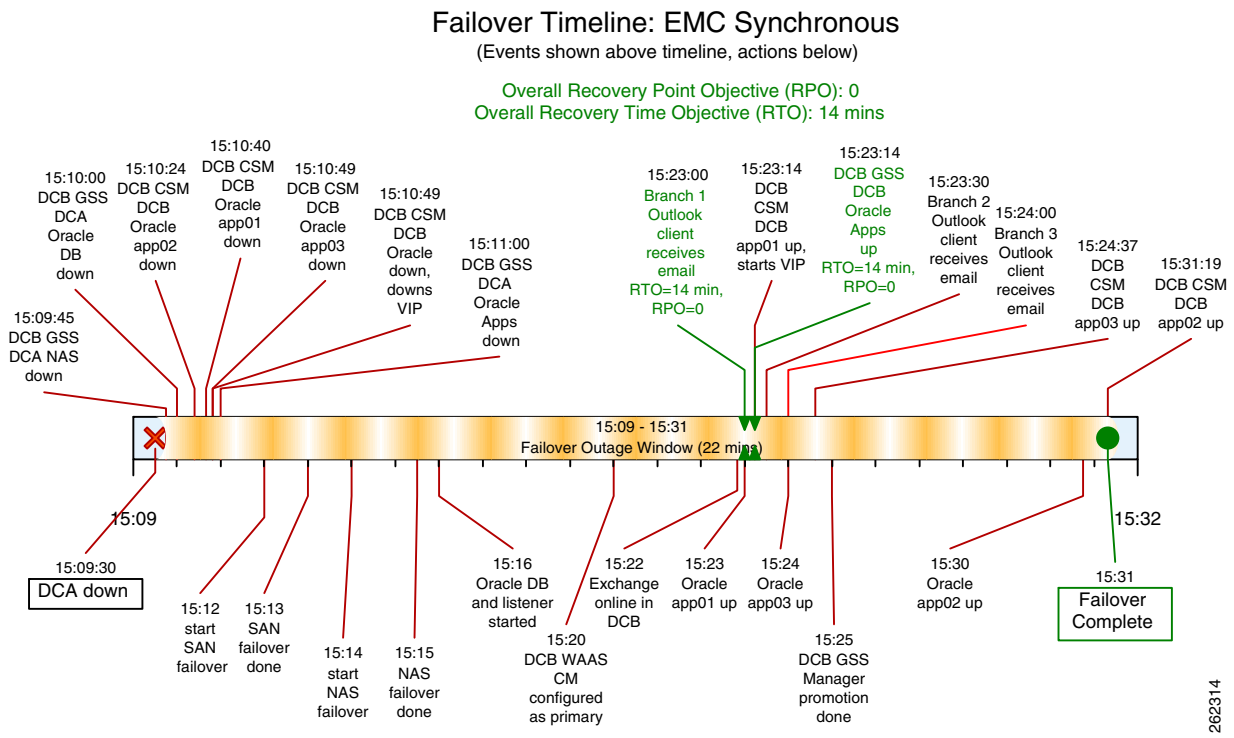


Figure 17-3 Failover Timeline—HP

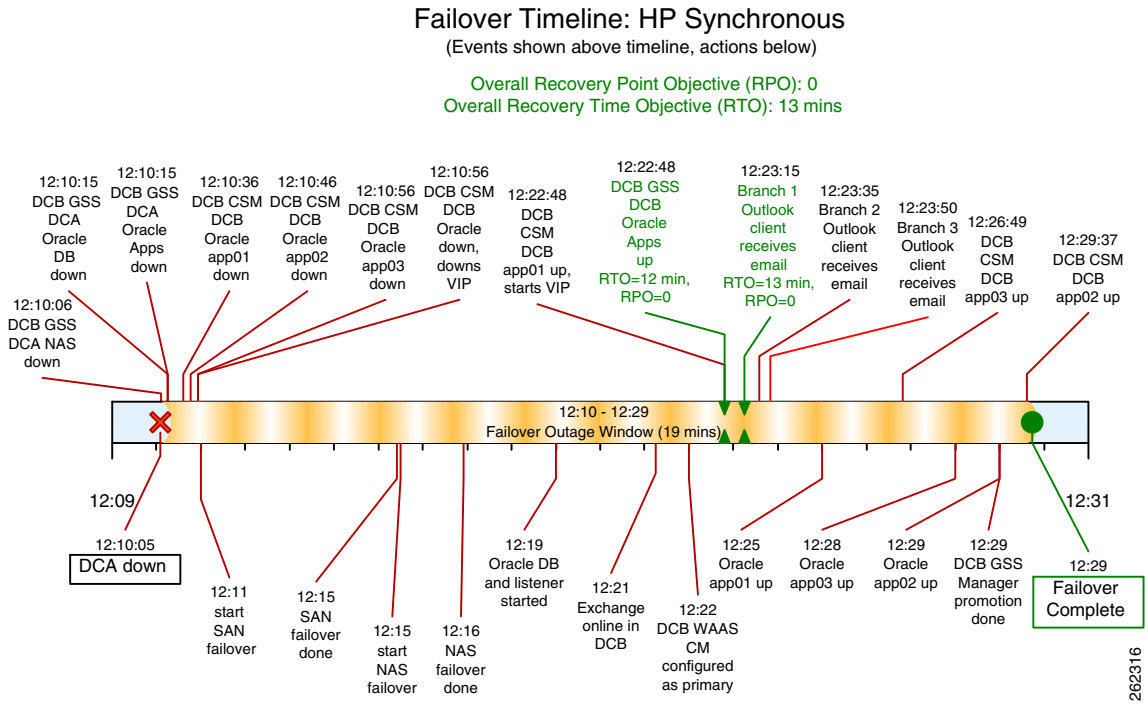


Figure 17-4 Failover Timeline—NetApp

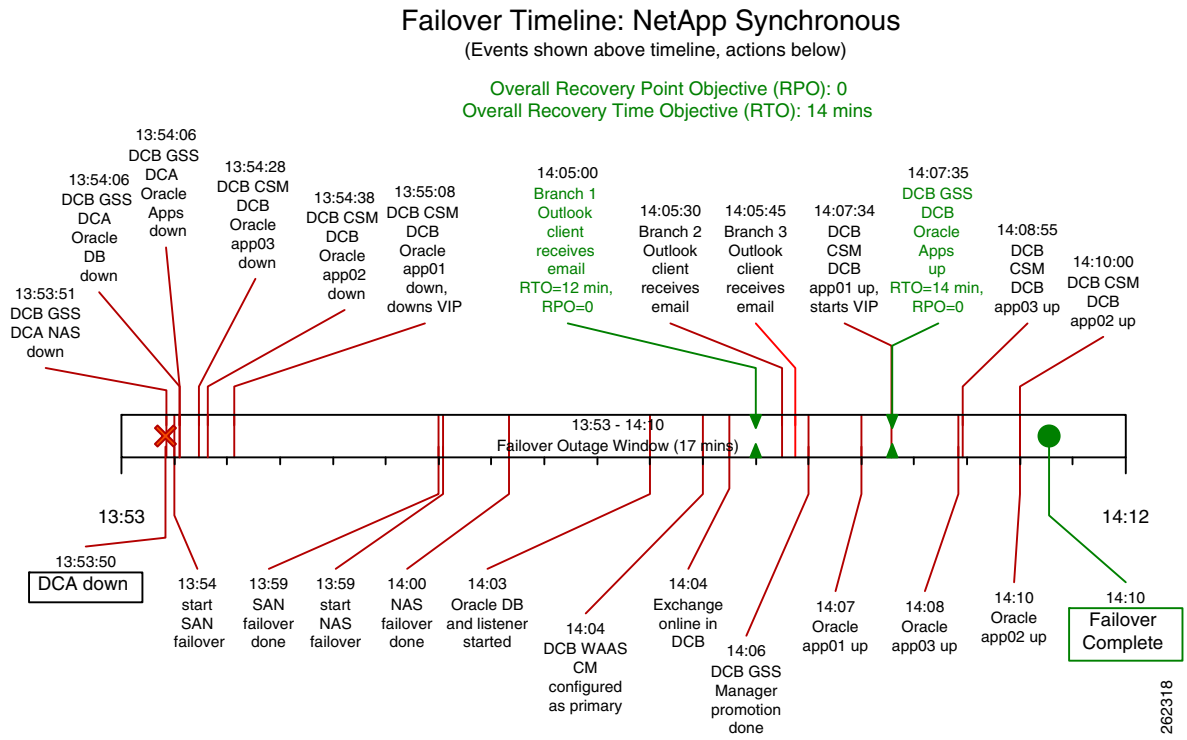


Figure 17-5 Failback Timeline—EMC

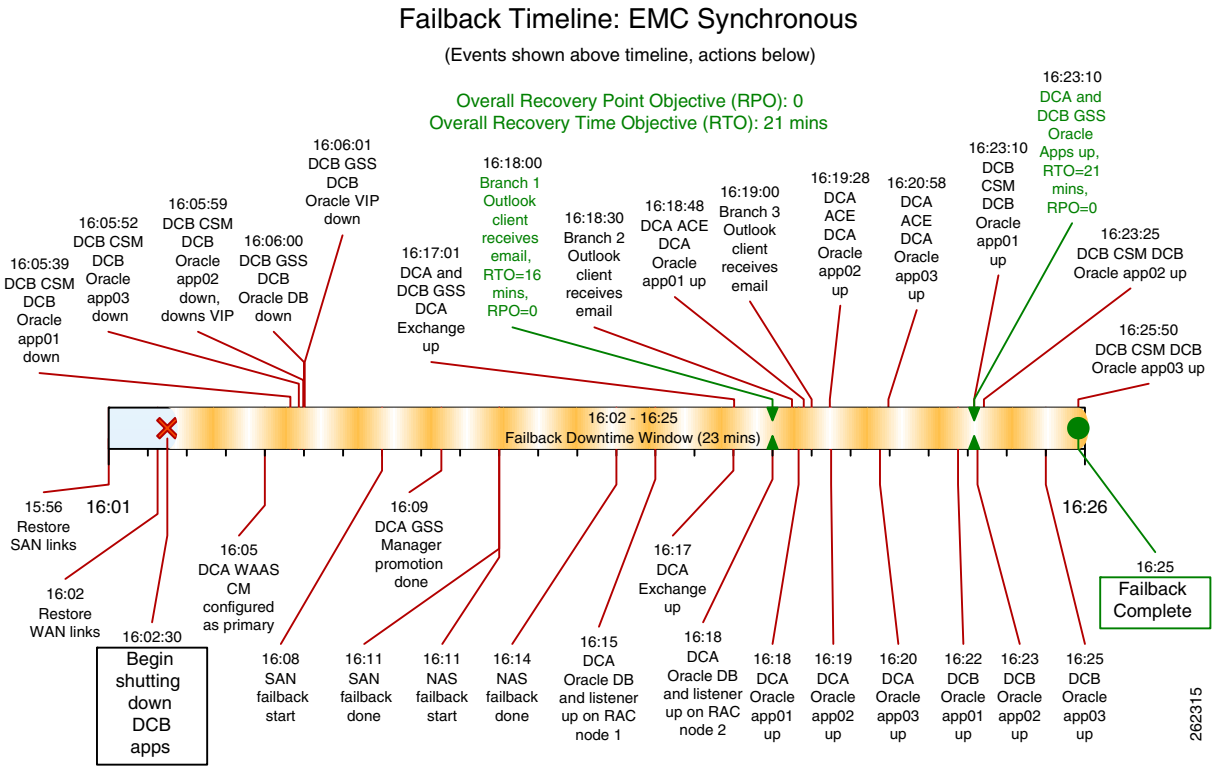


Figure 17-6 Failback Timeline—HP

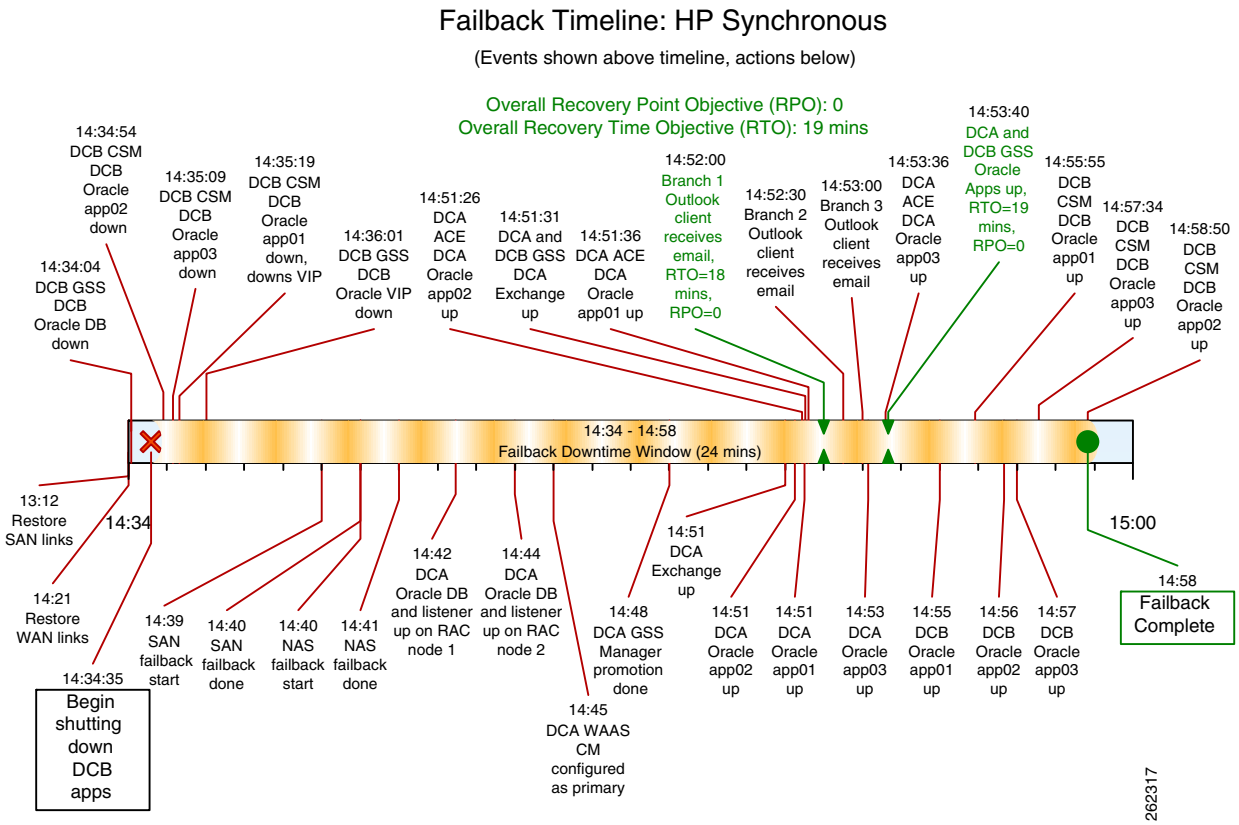
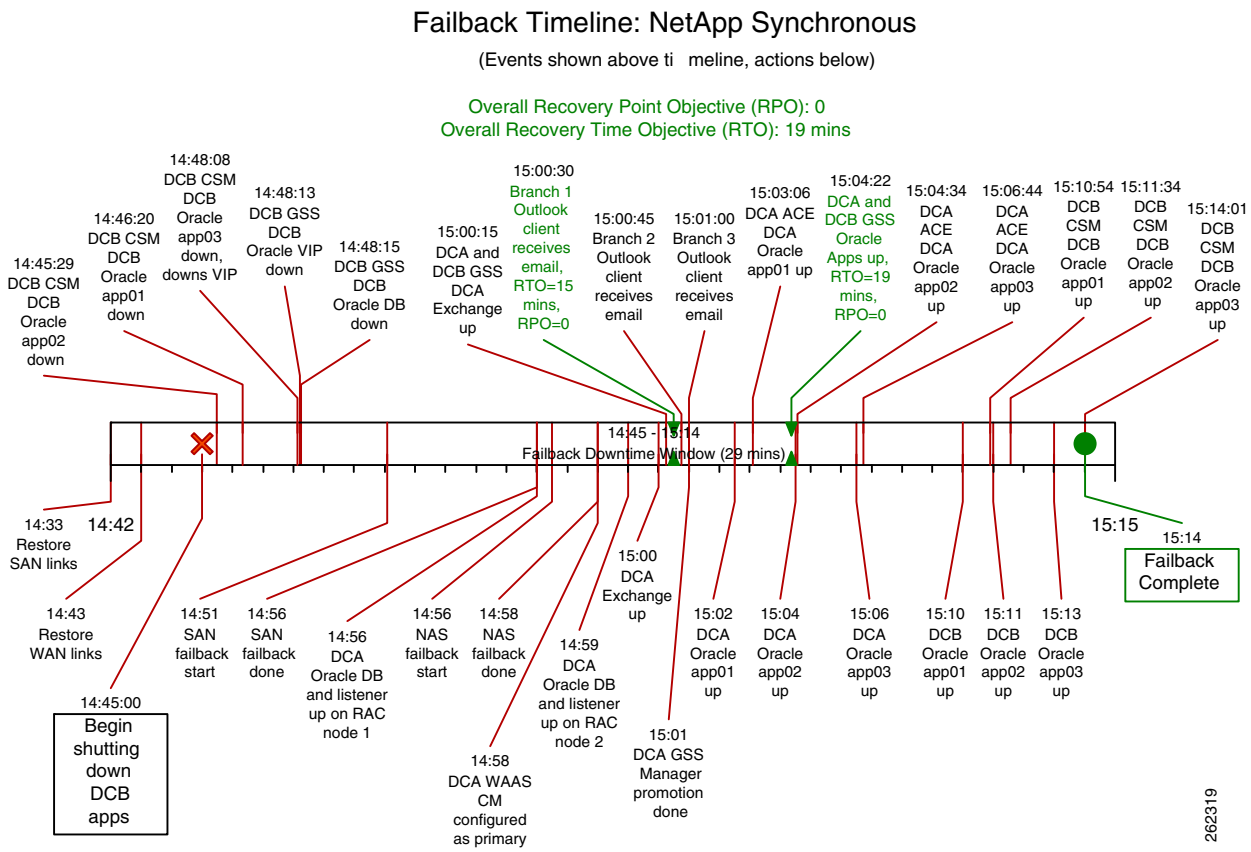


Figure 17-7 Failback Timeline—NetApp



For details about how to perform the failover and failback, refer to DCAP 4 Volume 12: Appendix.

High Availability Testing

The data center high availability tests include failing a component in the network and characterizing each application by the impact of that failure and the subsequent failover to a redundant component and failback to the primary component once it is again operational. All the information in the disaster recovery section about the topology and applications applies to high availability testing. In this phase, the key metrics in characterizing the application are the user experience and application throughput.

The user's experience during a component failure occurs in three major phases: failure, failover, and failback. The "failure" phase begins when the abnormal event occurs and ends when failover is complete. The "failover complete" phase begins when services have come up on a redundant device and ends when failback is initiated. The "failback initiated" phase begins when automatic or manual restoration of services on the original component begins and ends when the component has resumed its normal mode of operation. Although the "failback initiated" phase is optional in a real environment, in DCAP testing failback is done in every test.

As an example, Table 17-2 shows the three phases in the ACE graceful failover test.

Table 17-2 High Availability Phases in the ACE Graceful Failover Test

Phase	Start	End
Failure	manual failover is triggered with an **ft switchover** command	the standby ACE module is providing service
Failover Complete	the standby ACE module is providing service	manual failback is triggered with another **ft switchover** command
Failback Initiated	manual failback is triggered with another **ft switchover** command	the primary ACE module is providing service and the standby ACE is ready for another failover

Table 17-3 shows the user experience categories along with a key used in table A-4 to summarize the user experience for all the high availability tests.

Table 17-3 High Availability User Experience Categories

Key	User Experience Categories
	All user sessions on failed component impacted (application unavailable, redundant component not ready to provide service)
	Only existing sessions on failed component impacted (new sessions redirected to redundant component)
	No impact (normal)
N/A	Not applicable (failure scenario or component does not apply to application)

Table 17-4 characterizes the applications during high availability-related events by showing the user experience during each of the component failure phases. Table 17-4 also shows the overall impact of the entire event (all three phases) on application throughput. The Oracle throughput baseline number is the total number of user account records created by all Oracle E-Business Suite application users at all three branch locations accessing web and application servers in both data centers. The Exchange throughput baseline number is the total number of emails sent and received from Outlook 2003 users at all three branches to Exchange Virtual Servers in both data centers.

The throughput information for Oracle requires some explanation to aid in interpretation. The user load is generated by HP LoadRunner software, and the virtual user transaction timeout is set to 5 minutes due to some transactions taking nearly that long normally. When a disruption occurs due to a component failover, virtual users will wait up to 5 minutes before retrying a transaction. In contrast, a real user would have recognized the failure much sooner and initiated an attempt to start a new session which in most cases would succeed. As a result, the throughput measurement alone is not granular enough to indicate the true extent of disruption. Future improvements in testing and metric gathering will address this issue.

Table 17-4 High Availability Application Characterization

Category	Test Name	Application	User Experience			Application Throughput (% of baseline)
			Failure	Failover Complete	Failback Initiated	
ACE Mod	Graceful ACE Module Failover	Oracle				85%
		Exchange				100%
	Reset Primary ACE	Oracle				75%
		Exchange				100%
App Hosts	ESX Host Power Failure and Recovery-ACE Redirection	Oracle				91%
		Exchange	N/A	N/A	N/A	100%
	Graceful Host Shutdown and Recovery ACE Redirection	Oracle				93%
		Exchange	N/A	N/A	N/A	100%
	Physical Host Power Failure and Recovery-ACE Redirection	Oracle				52%
		Exchange	N/A	N/A	N/A	100%
	VM Application Host Power Failure and Recovery-ACE Redirection	Oracle				96%
		Exchange	N/A	N/A	N/A	100%
	VMotion Host-ACE Redirection	Oracle				100%
		Exchange	N/A	N/A	N/A	100%
	ESX Host Power Failure and Recovery-CSM Redirection	Oracle				86%
		Exchange				96%
	Graceful Host Shutdown and Recovery CSM Redirection	Oracle				89%
		Exchange	N/A	N/A	N/A	100%
	CSM-Physical Host Power Failure and Recovery	Oracle				89%
		Exchange	N/A	N/A	N/A	100%
	VM Host Power Failure and Recovery-CSM Redirection	Oracle				90%
		Exchange	N/A	N/A	N/A	98%
	VMotion Host-CSM Redirection	Oracle				89%
		Exchange	N/A	N/A	N/A	100%
	DC-NetApp Cluster Failover and Failback	Oracle				52%
		Exchange	N/A	N/A	N/A	100%
	DC-Primary Exchange Host Power Failure and Recovery	Oracle	N/A	N/A	N/A	96%
		Exchange				95%

Table 17-4 High Availability Application Characterization

CSM Module	Graceful CSM Module Failover	Oracle				100%
		Exchange	N/A	N/A	N/A	100%
	Power Cycle Primary CSM Module	Oracle				100%
		Exchange	N/A	N/A	N/A	100%
	Reset Primary CSM Module	Oracle				91%
		Exchange	N/A	N/A	N/A	100%
Device Failure	DCA Access 6k Reload	Oracle				93%
		Exchange				100%
	DCA Aggregation Switch Reload	Oracle				44%
		Exchange				83%
	DCA Core Reload	Oracle				100%
		Exchange				100%
	DCB Aggregation Switch Reload	Oracle				69%
		Exchange				99%
	GSS Interface Shutdown DCa	Oracle				85%
		Exchange				98%
	GSS Interface Shutdown DCb	Oracle				91%
		Exchange				100%
Link Failure	DCA Link Failures	Oracle				83%
		Exchange				100%
	DCB Link Failures	Oracle				96%
		Exchange				100%
SAN	HP c-Class BladeSystem MDS 9124e Reload and Recovery	Oracle				96%
		Exchange				100%
	MDS Core Switch Reload and Recovery	Oracle				91%
		Exchange				100%
	MDS Transit Switch Reload and Recovery	Oracle				96%
		Exchange				93%
WAAS	WAE Power Failure and Recovery ACE Redirection	Oracle				80%
		Exchange				88%
	WAE Standby Interface Failure and Recovery ACE Redirection	Oracle				98%
		Exchange				100%
	WAE Link Failure and Recovery WCCPv2 Redirection	Oracle				70%
		Exchange				95%
	WAE Power Failure and Recovery WCCPv2 Redirection	Oracle				81%
		Exchange				90%

Storage Replication Testing

The data center storage replication tests focus on storage replication mechanisms that can support a disaster recovery or business continuance strategy. The disaster recovery tests indirectly test storage replication, but this is limited to the synchronous fibre channel-based replication which is an inherent part of the active/active data center design. The data center storage replication tests focus on mechanisms which are not part of the data center design. In this phase, the only mechanism tested is NetApp SnapMirror.

The NetApp replication tests include SnapMirror in both synchronous and asynchronous modes over IP with and without optimization by WAAS. Replication was configured between a cluster of FAS6070s in DCa and a corresponding cluster of FAS6070s in DCb. A simulated distance of 2500 km was introduced by sending traffic through a WAN emulator with 40 ms of latency. RHEL 4 Linux hosts were presented both NAS and SAN storage which were populated with an assortment of Linux and Windows operating system files and Oracle and Exchange 2003 data files. For each test, these files were copied using the tar command, and the entire copy was timed. For synchronous tests, this was as simple as timing how long the tar command ran. For asynchronous tests, the time was the interval between when the tar command was initiated and the SnapMirror lag time was less than a minute (that is, all the residual data had been copied over to the target filer). After the copy was done, the storage was failed over and brought up on the host in the target data center, and the rsync command was used (with the "-n" or "dry run" switch, so that rsync itself didn't do any actual copying) to make sure the target data exactly matched the source data.

In the version of WAAS tested in this phase, there are known issues which affect SnapMirror throughput, and Table 17-5 shows the affects of these issues on throughput. Please see the individual tests for details on the known issues. These issues are addressed in more recent versions of WAAS.

Table 17-5 NetApp SnapMirror over IP with WAAS Testing Summary

	Throughput (KB/sec)		Speed Up	
Test	**SAN**	**NAS**	**SAN**	**NAS**
Asynchronous Replication over IP without WAAS	7413	7546	N/A	N/A
Asynchronous Replication over IP with WAAS	4038	2330	-46%	-69%
Synchronous Replication over IP without WAAS	261	718	N/A	N/A
Synchronous Replication over IP with WAAS	98	267	-62%	-63%

Test Results Summary

Table 17-6 summarizes results of all completed testing as part of the Cisco DCAP project for this release. Table 17-6 includes the feature or function tested, the section that describes the feature set to which the feature or function belongs, the results of the feature or function tests (pass, or pass with exception), the component tests for each feature or function, and any related defects found during testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested.

Refer to Volume 12: Appendix for a list of Cisco DCAP 4.0 DDTS's.

Table 17-6 DCAP Test Results Summary

Test Suites	Feature/Function	Tests	Results
Disaster Recovery, page 17-18	Fail Over, page 17-18	1. DC Disaster Recovery Failover EMC 2. DC Disaster Recovery Failover HP 3. DC Disaster Recovery Failover NetApp	
Disaster Recovery, page 17-18	Fail Back, page 17-23	1. DC Disaster Recovery Failback EMC 2. DC Disaster Recovery Failback HP 3. DC Disaster Recovery Failback NetApp	
High Availability, page 17-29	ACE Module, page 17-29	1. Graceful ACE Module Failover 2. Reset Primary ACE	
High Availability, page 17-29	Application Hosts, page 17-31	1. CSM-Physical Host Power Failure and Recovery 2. DC NetApp Cluster Failover and Failback 3. DC Primary Exchange Host Power Failure and Recovery 4. ESX Host Power Failure and Recovery ACE Redirection 5. ESX Host Power Failure and Recovery CSM Redirection 6. Graceful Host Shutdown and Recovery ACE Redirection 7. Graceful Host Shutdown and Recovery CSM Redirection 8. Physical Host Power Failure and Recovery- ACE Redirection 9. VM Application Host Power Failure and Recovery ACE Redirection 10. VM Host Power Failure and Recovery CSM Redirection 11. VMotion Host CSM Redirection 12. VMotion Host-ACE Redirection	
High Availability, page 17-29	Baseline, page 17-46	1. Generate Application Traffic for 15 Minutes	
High Availability, page 17-29	CSM Module, page 17-47	1. Graceful CSM Module Failover 2. Power Cycle Primary CSM Module 3. Reset Primary CSM Module	

Table 17-6 DCAP Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
High Availability, page 17-29	Device Failure, page 17-50	1. DCa Access 6k Reload 2. DCa Aggregation Switch Reload 3. DCa Core Reload 4. DCb Access 4k Reload 5. DCb Aggregation Switch Reload 6. GSS Interface Shutdown DCa 7. GSS Interface Shutdown DCb	
High Availability, page 17-29	Link Failure, page 17-56	1. DCa Link Failures 2. DCb Link Failures	
High Availability, page 17-29	SAN, page 17-59	1. HP c-Class BladeSystem MDS 9124e Reload and Recovery 2. MDS Core Switch Reload and Recovery 3. MDS Transit Switch Reload and Recovery	
High Availability, page 17-29	WAAS, page 17-62	1. WAE Link Failure and Recovery WCCPv2 Redirection 2. WAE Power Failure and Recovery ACE Redirection 3. WAE Power Failure and Recovery WCCPv2 Redirection 4. WAE Standby Interface Failure and Recovery ACE Redirection	CSCsl68531
Replication, page 17-67	NetApp, page 17-67	1. DC NetApp SnapMirror Async Disaster Recovery Replication over IP with WAAS 2. DC NetApp SnapMirror Async Disaster Recovery Replication over IP without WAAS 3. DC NetApp SnapMirror Sync Disaster Recovery Replication over IP with WAAS 4. DC NetApp SnapMirror Sync Disaster Recovery Replication over IP without WAAS	CSCsh72271 CSCsg79439 , CSCsh72271 , CSCsg79439 ,

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

- Disaster Recovery, page 17-18
- High Availability, page 17-29
- Replication, page 17-67

Disaster Recovery

The disaster recovery tests ensure the DCAP dual data center topology is properly configured to support failover of the Oracle and Microsoft Exchange applications to data center B and failback to data center A. A simulated distance of 100 km was in effect for all tests.

This section contains the following topics:

- Fail Over, page 17-18
- Fail Back, page 17-23

Fail Over

Three separate tests, one for each storage vendor, are performed. The tests consist of application clients initiating transactions continuously from each of the three branch locations. After a short time, a failure of data center A is simulated by severing all its WAN and SAN replication links. After the failover for each application is done, the data is checked to make sure data from all transactions completed prior to the point of the failure is available in data center B. This determines the recovery point objective (RPO). The recovery time objective (RTO) is determined by measuring the time from the failure to the time when the last application becomes available to at least one client in data center B.

This section contains the following topics:

- DC Disaster Recovery Failover EMC, page 17-18
- DC Disaster Recovery Failover HP, page 17-20
- DC Disaster Recovery Failover NetApp, page 17-21

DC Disaster Recovery Failover EMC

This test verified that Oracle E-business Suite 11i, Oracle database 10gR2 RAC, and Exchange 2003 Server failed over as expected in a disaster recovery scenario. Prior to the failover Oracle Applications is configured to run in an active-active hybrid mode (Application Layer active in both Data Centers and Database is Active in only one Data Center). Exchange is configured to run in an active-standby mode (back end only). Storage and application failover procedures are triggered manually, since both the Oracle and Exchange primary and failover host clusters are local to each data center. Oracle Application NAS storage is replicated synchronously using synchronous SnapMirror over IP and accelerated by WAAS and WAFS. MDS switches provide fiber channel SAN connectivity and replication services for Oracle and Exchange database servers. Oracle and Exchange SAN storage is replicated synchronously using EMC's SRDF/S over a simulated 100 km distance. Key metrics gathered during testing are

Recovery Time Objective or RTO (time it takes to fail over each application and the data center as a whole) and Recovery Point Objective or RPO (amount of data lost due to the disaster; should be none due to use of synchronous replication).

Test Procedure

The procedure used to perform the DC Disaster Recovery Failover EMC test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Oracle: Begin sending Load Runner traffic and verify the application and network is functioning properly. Exchange: Verify Exchange application is running normally, then begin sending email from one Linux host in each DC and verify email reception for each branch server Outlook client (branch 1: testuser.00001, testuser.00004; branch 2: testuser.00002, testuser.00005; branch 3: testuser.00003, testuser.00006). Verify GSS. Verify CSM. Verify ACE. Verify Load Runner. Verify WAAS. Verify NAS and SAN storage replication.

Step 3 Simulate a disaster situation in which all connectivity to DCA is terminated. Note the time.

Step 4 Fail over Oracle (Database) and Exchange SAN storage.

Step 5 Fail over Oracle (Applications) NAS storage.

Step 6 Bring up Exchange database on the failover cluster and verify all branch clients can receive email. Note the time (this is the Exchange Recovery Point Objective or RPO). Also verify how much data (email) if any, was lost (this is the Exchange Recovery Time Objective or RTO). Should be no data loss.

Step 7 Bring up Oracle database on the failover cluster.

Step 8 Activate WAAS Central Manager in DCB.

Step 9 Verify GSS is giving out correct addresses for Oracle database, NAS, and clients. Then remount NAS storage on Oracle client nodes (may require reboot).

Step 10 Bring up Oracle application on the failover nodes, verify CSM, and verify GSS is directing all clients to DCB. Note the time (this is the Oracle Recovery Time Objective or RTO). Also verify how much data, if any, was lost (this is the Oracle Recovery Point Objective or RPO). Should be no data loss.

Step 11 Stop any scripts that might still be running (LoadRunner, email, MDS switch counters, SAN replication checks).

Step 12 Determine the latest RTO of all applications. This is the data center failover RTO. Determine the earliest RPO of all applications. This is the data center failover RPO.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that SAN replication and fail over will be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect that CSM will automatically load balance Oracle clients to both Oracle application servers in DCB after failover.

- We expect that GSS will automatically direct Oracle clients to a sorry server during failover and then direct all clients to DCB after failover.
- We expect that all GSS will automatically direct NAS clients to the DCB NAS filer after failover.
- We expect that GSS will automatically direct Oracle database clients to the DCB Oracle server after failover.
- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all applications will have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

DC Disaster Recovery Failover EMC passed.

DC Disaster Recovery Failover HP

This test verified that Oracle E-business Suite 11i, Oracle database 10gR2, and Exchange 2003 Server failed over as expected in a disaster recovery scenario. Prior to the failover Oracle Applications is configured to run in an active-active hybrid mode (Application Layer active in both Data Centers and Database is Active in only one Data Center). Exchange is configured to run in an active-standby mode (back end only). Storage and application failover procedures are triggered manually, since both the Oracle and Exchange primary and failover host clusters are local to each data center. Oracle Application NAS storage is replicated synchronously using synchronous snapmirror over IP and accelerated by WAAS and WAFS. MDS switches provide fiber channel SAN connectivity and replication services for Oracle and Exchange database servers. Oracle and Exchange SAN storage is replicated synchronously using HP's Continuous Access XP Sync over a simulated 100 km distance. Key metrics gathered during testing are Recovery Time Objective or RTO (time it takes to fail over each application and the data center as a whole) and Recovery Point Objective or RPO (amount of data lost due to the disaster; should be none due to use of synchronous replication).

Test Procedure

The procedure used to perform the DC Disaster Recovery Failover HP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Follow these steps:

```
Oracle: Begin sending Load Runner traffic and verify the application and network is
functioning properly.
Exchange: Verify Exchange application is running normally, then begin sending email from
one Linux host in each DC and verify email reception for each branch server Outlook client
(branch 1: testuser.00001, branch 2: testuser.00002, branch 3: testuser.00003).
Verify GSS.
Verify ACE.
Verify CSM.
Verify Load Runner
Verify WAAS
Verify NAS and SAN storage replication
```

Step 3 Simulate a disaster situation in which all connectivity to DCA is terminated. Note the time.

Step 4 Fail over Oracle (Database) and Exchange SAN storage.

Step 5 Fail over Oracle (Applications) NAS storage.

Step 6 Bring up Exchange database on the failover cluster and verify all branch clients can receive email. Note the time (this is the Exchange Recovery Point Objective or RPO). Also verify how much data (email) if any, was lost (this is the Exchange Recovery Time Objective or RTO). Should be no data loss.

Step 7 Bring up Oracle database on the failover cluster.

Step 8 Verify GSS is giving out correct addresses for Oracle database, NAS, and clients. Then remount NAS storage on Oracle client nodes (may require reboot).

Step 9 Bring up Oracle application on the failover nodes, verify CSM, and verify GSS is directing all clients to DCB. Note the time (this is the Oracle Recovery Time Objective or RTO). Also verify how much data, if any, was lost (this is the Oracle Recovery Point Objective or RPO). Should be no data loss.

Step 10 Stop any scripts that might still be running (LoadRunner, email, MDS switch counters, SAN replication checks).

Step 11 Determine the latest RTO of all applications. This is the data center failover RTO. Determine the earliest RPO of all applications. This is the data center failover RPO.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that SAN replication and fail over will be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect that CSM will automatically load balance Oracle clients to both Oracle application servers in DCB after failover.
- We expect that GSS will automatically direct Oracle clients to a sorry server during failover and then direct all clients to DCB after failover.
- We expect that all GSS will automatically direct NAS clients to the DCB NAS filer after failover.
- We expect that GSS will automatically direct Oracle database clients to the DCB Oracle server after failover.
- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all applications will have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

DC Disaster Recovery Failover HP passed.

DC Disaster Recovery Failover NetApp

This test verified that Oracle E-business Suite 11i, Oracle database 10gR2, and Exchange 2003 Server failed over as expected in a disaster recovery scenario. Prior to the failover Oracle Applications is configured to run in an active-active hybrid mode (Application Layer active in both Data Centers and

Database is Active in only one Data Center). Exchange is configured to run in an active-standby mode (back end only). Storage and application failover procedures are triggered manually, since both the Oracle and Exchange primary and failover host clusters are local to each data center. Oracle Application NAS storage is replicated synchronously using synchronous snapmirror over IP and accelerated by WAAS and WAFS. MDS switches provide fiber channel SAN connectivity and replication services for Oracle and Exchange database servers. Oracle and Exchange SAN storage is replicated synchronously using NetApp's synchronous SnapMirror over a simulated 100 km distance. Key metrics gathered during testing are Recovery Time Objective or RTO (time it takes to fail over each application and the data center as a whole) and Recovery Point Objective or RPO (amount of data lost due to the disaster; should be none due to use of synchronous replication).

Test Procedure

The procedure used to perform the DC Disaster Recovery Failover NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Follow these steps:

```
Oracle: Begin sending Load Runner traffic and verify the application and network is
functioning properly.
Exchange: Verify Exchange application is running normally, then begin sending email from
one Linux host in each DC and verify email reception for each branch server Outlook client
(branch 1: testuser.00001, branch 2: testuser.00002, branch 3: testuser.00003).
Verify GSS.
Verify ACE.
Verify CSM.
Verify Load Runner.
Verify WAAS.
Verify NAS and SAN storage replication
```

Step 3 Simulate a disaster situation in which all connectivity to DCA is terminated. Note the time.

Step 4 Fail over Oracle (Database) and Exchange SAN storage.

Step 5 Fail over Oracle (Applications) NAS storage.

Step 6 Bring up Exchange database on the failover cluster and verify all branch clients can receive email. Note the time (this is the Exchange Recovery Point Objective or RPO). Also verify how much data (email) if any, was lost (this is the Exchange Recovery Time Objective or RTO). Should be no data loss.

Step 7 Bring up Oracle database on the failover cluster.

Step 8 Verify GSS is giving out correct addresses for Oracle database, NAS, and clients. Then remount NAS storage on Oracle client nodes (may require reboot).

Step 9 Bring up Oracle application on the failover nodes, verify CSM, and verify GSS is directing all clients to DCB. Note the time (this is the Oracle Recovery Time Objective or RTO). Also verify how much data, if any, was lost (this is the Oracle Recovery Point Objective or RPO). Should be no data loss.

Step 10 Stop any scripts that might still be running (LoadRunner, email, MDS switch counters, SAN replication checks).

Step 11 Determine the latest RTO of all applications. This is the data center failover RTO. Determine the earliest RPO of all applications. This is the data center failover RPO.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that SAN replication and fail over will be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect that CSM will automatically load balance Oracle clients to both Oracle application servers in DCB after failover.
- We expect that GSS will automatically direct Oracle clients to a sorry server during failover and then direct all clients to DCB after failover.
- We expect that all GSS will automatically direct NAS clients to the DCB NAS filer after failover.
- We expect that GSS will automatically direct Oracle database clients to the DCB Oracle server after failover.
- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all applications will have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

DC Disaster Recovery Failover NetApp passed.

Fail Back

Three separate tests, one for each storage vendor, are performed. The tests essentially reverse the effect of the fail over test, except rather than a simulated failure being the first step, a controlled shutdown of applications starts the test. After the failback for each application is done, the data is checked to make sure data from all transactions completed prior to the start of the shutdown is available in data center A. This determines the recovery point objective (RPO). The recovery time objective (RTO) is determined by measuring the time from the shutdown to the time when the last application becomes available to at least one client in data center A.

This section contains the following topics:

DC Disaster Recovery Failback EMC

This test verified that Oracle E-business Suite 11i, Oracle Database 10gR2 RAC, and Exchange 2003 Server failed back as expected in a disaster recovery scenario. Prior to the failback Oracle and Exchange are running in the failover data center. The assumption is that the same storage devices are available, i.e. the simulated disaster did not destroy them or the data center itself. A further assumption is that failback occurs during a scheduled downtime, so application utilization is low. Storage and application failback procedures are triggered manually, since both the Oracle Database and Exchange primary and failover

host clusters are local to each data center. GSS, CSM, and WAAS procedures are automatic (except for designating a new management node for GSS and WAAS). Oracle Applications NAS storage is replicated synchronously back to the primary data center using synchronous snapmirror over IP and accelerated by WAAS and WAFS. MDS switches provide fiber channel SAN connectivity and replication services for Oracle and Exchange database servers. Oracle (Database) and Exchange SAN storage is replicated synchronously back to the primary data center using EMC's SRDF/S over a simulated 100 km distance. Key metrics gathered during testing are Recovery Time Objective or RTO (time it takes to fail back each application and the data center as a whole) and Recovery Point Objective or RPO (amount of data lost due to the fail back; should be none due to use of synchronous replication). After failback is complete, storage replication is re-enabled to put both data centers back in normal disaster preparedness mode.

Test Procedure

The procedure used to perform the DC Disaster Recovery Failback EMC test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Oracle: Begin sending Load Runner traffic and verify the application and network is functioning properly. Exchange: Verify Exchange application is running normally, then begin sending email from one Linux host in each DC and verify email reception for each branch server Outlook client (branch 1: testuser.00001, testuser.00004; branch 2: testuser.00002, testuser.00005; branch 3: testuser.00003, testuser.00006). Verify GSS. Verify ACE. Verify CSM. Verify Load Runner. Verify WAAS. Verify NAS and SAN storage replication.

Step 3 Ensure the primary data center storage array is in the proper state, then restore SAN extension connectivity to the primary data center. As appropriate, begin resynchronization of the failover data center storage back to the primary data center (only if application downtime is not required.)

Step 4 Ensure the primary data center applications, including the ACE VIP for Oracle, are offline, then restore WAN connectivity to DCA.

Step 5 After the failback outage window begins, ensure all applications are offline in the failover data center, then fail back SAN storage.

Step 6 Fail back Oracle Applications NAS storage.

Step 7 Bring up Exchange database on the primary cluster and note the time when the first branch client can receive email (this is the Exchange Recovery Time Objective (or RTO). Also note the time when all clients can receive email. Also verify how much data (email) if any, was lost. This is the Exchange Recovery Point Objective (or RPO). Should be no data loss.

Step 8 Bring up Oracle database and the DB Listener on the primary cluster.

Step 9 Activate WAAS Central Manager in DCA.

Step 10 Verify GSS is giving out correct addresses for Oracle database, NAS, and clients. Then remount NAS storage on all Oracle application nodes (may require reboot).

Step 11 Bring up Oracle application on the all Application nodes in both Data Centers, verify CSM, and verify GSS is loadbalancing clients to both DCA and DCB. Note the time when the first branch client can access Oracle (this is the Oracle Recovery Time Objective (or RTO). Also note the time when all clients can access Oracle. Also verify how much data, if any, was lost. This is the Oracle Recovery Point Objective (or RPO). Should be no data loss.

Step 12 Stop any scripts that might still be running (LoadRunner, email, MDS switch counters, SAN replication checks).

Step 13 Reinstate DCA to DCB replication for both SAN and NAS storage.

Step 14 Determine the latest RTO of all applications. This is the data center failback RTO. Determine the earliest RPO of all applications. This is the data center failback RPO.

Step 15 Stop background scripts to collect final status of network devices and analyze for error.

Step 16 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that SAN replication and failback will be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect that ACE (in DCa) and CSM (in DCb) will automatically load balance Oracle clients to both Oracle application servers in both data centers after failback.
- We expect that GSS will automatically direct Oracle clients to a sorry server during failback and then direct all clients to both data centers using the configured load balancing metric after failback.
- We expect that GSS will automatically direct NAS clients to the DCA NAS filer after failback.
- We expect that GSS will automatically direct Oracle database clients to the DCA Oracle server after failback.
- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all applications will have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

DC Disaster Recovery Failback EMC passed.

DC Disaster Recovery Failback HP

This test verified that Oracle E-business Suite 11i, Oracle Database 10gR2, and Exchange 2003 Server failed back as expected in a disaster recovery scenario. Prior to the failback Oracle and Exchange are running in the failover data center. The assumption is that the same storage devices are available, i.e. the simulated disaster did not destroy them or the data center itself. A further assumption is that failback occurs during a scheduled downtime, so application utilization is low. Storage and application failback procedures are triggered manually, since both the Oracle Database and Exchange primary and failover host clusters are local to each data center. GSS, CSM, and WAAS procedures are automatic (except for designating a new management node for GSS and WAAS). Oracle Applications NAS storage is replicated synchronously back to the primary data center using synchronous snapmirror over IP and accelerated by WAAS and WAFS. MDS switches provide fiber channel SAN connectivity and replication services for Oracle and Exchange database servers. Oracle (Database) and Exchange SAN storage is replicated synchronously back to the primary data center using HP's Continuous Access XP Sync over a simulated 100 km distance. Key metrics gathered during testing are Recovery Time Objective or RTO (time it takes to fail back each application and the data center as a whole) and Recovery Point Objective or RPO (amount of data lost due to the fail back; should be none due to use of synchronous replication). After failback is complete, storage replication is re-enabled to put both data centers back in normal disaster preparedness mode.

Test Procedure

The procedure used to perform the DC Disaster Recovery Failback HP test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Follow these steps:

```
Oracle: Begin sending Load Runner traffic and verify the application and network is
functioning properly.
Exchange: Verify Exchange application is running normally, then begin sending email from
one Linux host in each DC and verify email reception for each branch server Outlook client
(branch 1: testuser.00001, branch 2: testuser.00002, branch 3: testuser.00003).
Verify ACE
Verify GSS
Verify CSM
Verify Load Runner
Verify WAAS
Verify NAS and SAN storage replication
```

Step 3 Ensure the primary data center storage array is in the proper state, then restore SAN extension connectivity to the primary data center. As appropriate, begin resynchronization of the failover data center storage back to the primary data center (only if application downtime is not required.)

Step 4 Ensure the primary data center applications, including the ACE VIP for Oracle, are offline, then restore WAN connectivity to DCA.

Step 5 After the failback outage window begins, ensure all applications are offline in the failover data center, then fail back SAN storage.

Step 6 Fail back Oracle Applications NAS storage.

Step 7 Bring up Exchange database on the primary cluster and note the time when the first branch client can receive email (this is the Exchange Recovery Time Objective or RTO). Also note the time when all clients can receive email. Also verify how much data (email) if any, was lost (this is the Exchange Recovery Point Objective or RPO). Should be no data loss.

Step 8 Bring up Oracle database and the DB Listener on the primary cluster.

Step 9 Verify GSS is giving out correct addresses for Oracle database, NAS, and clients. Then remount NAS storage on all Oracle client nodes (should not require reboot).

Step 10 Bring up Oracle application on the all Application nodes in both Data Centers, verify CSM, and verify GSS is loadbalancing clients to both DCA and DCB. Note the time (this is the Oracle Recovery Time Objective or RTO). Also verify how much data, if any, was lost (this is the Oracle Recovery Point Objective or RPO). Should be no data loss.

Step 11 Stop any scripts that might still be running (LoadRunner, email, MDS switch counters, SAN replication checks).

Step 12 Reinstate DCA to DCB replication for both SAN and NAS storage.

Step 13 Determine the latest RTO of all applications. This is the data center failback RTO. Determine the earliest RPO of all applications. This is the data center failback RPO.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that SAN replication and failback will be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect that all CSM will automatically load balance Oracle clients to both Oracle application servers in both data centers after failback.
- We expect that GSS will automatically direct Oracle clients to a sorry server during failback and then direct all clients to both data centers using the configured load balancing metric after failback.
- We expect that GSS will automatically direct NAS clients to the DCA NAS filer after failback.
- We expect that GSS will automatically direct Oracle database clients to the DCA Oracle server after failback.
- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all applications will have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

DC Disaster Recovery Failback HP passed.

DC Disaster Recovery Failback NetApp

This test verified that Oracle E-business Suite 11i, Oracle Database 10gR2, and Exchange 2003 Server failed back as expected in a disaster recovery scenario. Prior to the failback Oracle and Exchange are running in the failover data center. The assumption is that the same storage devices are available, i.e. the simulated disaster did not destroy them or the data center itself. A further assumption is that failback occurs during a scheduled downtime, so application utilization is low. Storage and application failback procedures are triggered manually, since both the Oracle Database and Exchange primary and failover host clusters are local to each data center. GSS, CSM, and WAAS procedures are automatic (except for designating a new management node for GSS and WAAS). Oracle Applications NAS storage is replicated synchronously back to the primary data center using synchronous snapmirror over IP and accelerated by WAAS and WAFS. MDS switches provide fiber channel SAN connectivity and replication services for Oracle and Exchange database servers. Oracle (Database) and Exchange SAN storage is replicated synchronously back to the primary data center using NetApp's synchronous SnapMirror over a simulated 100 km distance. Key metrics gathered during testing are Recovery Time Objective or RTO (time it takes to fail back each application and the data center as a whole) and Recovery Point Objective or RPO (amount of data lost due to the fail back; should be none due to use of synchronous replication). After failback is complete, storage replication is re-enabled to put both data centers back in normal disaster preparedness mode.

Test Procedure

The procedure used to perform the DC Disaster Recovery Failback NetApp test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Oracle: Begin sending Load Runner traffic and verify the application and network is functioning properly. Exchange: Verify Exchange application is running normally, then begin sending email from one Linux host in each DC and verify email reception for each branch server Outlook client (branch 1: testuser.00001, branch 2: testuser.00002, branch 3: testuser.00003). Verify GSS Verify CSM Verify Load Runner Verify WAAS Verify NAS and SAN storage replication

Step 3 Ensure the primary data center storage array is in the proper state, then restore SAN extension connectivity to the primary data center. As appropriate, begin resynchronization of the failover data center storage back to the primary data center (only if application downtime is not required.)

Step 4 Ensure the primary data center applications, including the ACE VIP for Oracle, are offline, then restore WAN connectivity to DCA.

Step 5 After the failback outage window begins, ensure all applications are offline in the failover data center, then fail back SAN storage.

Step 6 Fail back Oracle Applications NAS storage.

Step 7 Bring up Exchange database on the primary cluster and note the time when the first branch client can receive email (this is the Exchange Recovery Time Objective or RTO). Also note the time when all clients can receive email. Also verify how much data (email) if any, was lost (this is the Exchange Recovery Point Objective or RPO). Should be no data loss.

Step 8 Bring up Oracle database and the DB Listener on the primary cluster.

Step 9 Verify GSS is giving out correct addresses for Oracle database, NAS, and clients. Then remount NAS storage on all Oracle client nodes (may require reboot).

Step 10 Bring up Oracle application on the all Application nodes in both Data Centers, verify CSM, and verify GSS is loadbalancing clients to both DCA and DCB. Note the time (this is the Oracle Recovery Time Objective or RTO). Also verify how much data, if any, was lost (this is the Oracle Recovery Point Objective or RPO). Should be no data loss.

Step 11 Stop any scripts that might still be running (LoadRunner, email, MDS switch counters, SAN replication checks).

Step 12 Reinstate DCA to DCB replication for both SAN and NAS storage.

Step 13 Determine the latest RTO of all applications. This is the data center failback RTO. Determine the earliest RPO of all applications. This is the data center failback RPO.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that SAN replication and failback will be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect that all CSM will automatically load balance Oracle clients to both Oracle application servers in both data centers after failback.
- We expect that GSS will automatically direct Oracle clients to a sorry server during failback and then direct all clients to both data centers using the configured load balancing metric after failback.
- We expect that GSS will automatically direct NAS clients to the DCA NAS filer after failback.
- We expect that GSS will automatically direct Oracle database clients to the DCA Oracle server after failback.

- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all applications will have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

DC Disaster Recovery Failback NetApp passed.

High Availability

High availability is a system design protocol and associated implementation that ensures a certain absolute degree of operational continuity during a given measurement period.

This section contains the following topics:

ACE Module

This suite of tests focused on high availability scenarios focusing on the ACE module.

This section contains the following topics:

Graceful ACE Module Failover

This test verified the functionality of the Application(s) during primary ACE FT switchover. This involved sending DCAP application traffic from each branch, forcing an ACE module FT switchover and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. On the ACE preempt was unconfigured so that a graceful ft switchover could be done. It was verified that the standby ACE became active. Once the standby ACE became active we gracefully failed back to the intial active ACE.

Test Procedure

The procedure used to perform the Graceful ACE Module Failover test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 To perform a graceful ft failover on the ACE preempt must first be unconfigured by issuing the **no preempt** command on both the active and standby ACE.

Step 4 Initiate the test traffic for a duration of 10 minutes.

Step 5 Verify the state of the ACE modules by issuing the **show ft group summary** command.

Step 6 Perform the component failure by issuing the **ft switchover** command.

Step 7 Verify the state of the new active ACE by issuing the **show ft group summary** command.

Step 8 Perform a graceful switchover to the initial active ACE by issuing the **ft switchover** command on the new active ACE.

Step 9 Verify the initially active ACE once again is active by issuing the **show ft group summary** command. Also verify the initial standby ACE is again in the STANDBY_HOT state.

Step 10 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 11 Reconfigure preempt on both the active and standby ACE by issuing the **preempt** command.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure has occurred.
- We expect minimal impact to the end user. The potential for short lived HTTP flows to fail during failover/failback.
- We expect the standby ACE to become primary during power cycle and revert to standby after original primary comes active.

Results

Graceful ACE Module Failover passed.

Reset Primary ACE

This test verified the functionality of the Application(s) during primary ACE module failure. This involved sending DCAP application traffic from each branch, power cycling ACE module and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The primary ACE was power cycled and it was verified that the standby ACE became active. Once the power cycled ACE comes back online it should preempt and once again become the active ACE.

Test Procedure

The procedure used to perform the Reset Primary ACE test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify that preempt is enabled on both ACE modules by issuing the **show ft group summary** command.

Step 5 Verify the state of the ACE modules by issuing the **show ft group summary** command.

Step 6 Perform the component failure by issuing the **hw-module module *module*** reset command on dca-agg-1.

Step 7 Verify the state of the new active ACE by issuing the **show ft group summary** command.

Step 8 Verify the initially active ACE comes back online and preempts, once again becoming the active ACE by issuing the **show ft group summary** command. Also verify the initial standby ACE is again in the STANDBY_HOT state.

Step 9 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure has occurred.
- We expect minimal impact to the end user. The potential for short lived HTTP flows to fail during failover/failback.
- We expect the standby ACE to become primary during power cycle and revert to standby after original primary comes active.

Results

Reset Primary ACE passed.

Application Hosts

This suite of tests focused on high availability scenarios focusing on virtual and physical application hosts.

This section contains the following topics:

CSM-Physical Host Power Failure and Recovery

This test verified the functionality of the Application(s) during a Physical application host failures. This involved sending DCAP application traffic from each branch to both Data Centers, power cycling an application host and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and CSM load balances connections in a round robin fashion. One of the physical blade server hosts is power cycled and it was verified that the existing connections to the application host were terminated and after a probe failure by CSM. Due to this probe failure the CSM takes the host out of service. All the new connections will be sent to the other application hosts until the power cycled host recovers. When host comes back online it was verified that CSM brings the physical host back in service and accepts new connections./p>

Test Procedure

The procedure used to perform the CSM-Physical Host Power Failure and Recovery test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total user connections to the application host from CSM by issuing the **show mod csm 2 serverfarms name oracle-all detail** command.

Step 5 Perform the component failure by powering down the application blade host via the HP Onboard Administrator for at least 2 minutes.

Step 6 Verify the CSM has brought the host out of service after the component failure by issuing the **show mod csm 2 probe name oracle detail** command on the CSM.

Step 7 Verify all new connections are redirected to the other application hosts by issuing the **show mod csm 2 serverfarms name oracle-all detail** command.

Step 8 Power on the application host in the HP Onboard Adminstrator GUI.

Step 9 When the host comes back online, start the application services by issuing the **./adstrtal.sh apps/apps** command.

Step 10 Verify that the CSM brings the host back inservice once the application services are up by issuing the **show mod csm 2 probe name oracle detail** command.

Step 11 Verify the CSM is again sending requests to the host by issuing the **show mod csm 2 serverfarms name oracle-all detail** command.

Step 12 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing application traffic connected to the physical host to fail.
- We expect new connections to continue to pass after the component failure has occurred.
- We expect the CSM to recognize the physical host recovery and bring it back in service to accept new connections.

Results

CSM-Physical Host Power Failure and Recovery passed.

DC NetApp Cluster Failover and Failback

This test verified the functionality of the application(s) during a NetApp cluster failover and failback. This involved sending DCAP application traffic from each branch to both Data Centers, rebooting the active NetApp cluster node in the primary data center, doing a cluster giveback when it came back up, and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The Oracle traffic was load balanced across 3 application hosts and ACE and CSM load balanced connections in a round robin fashion. The primary NetApp filer providing NAS storage was failed over and failed back, and it was verified that ACE and CSM reported probe failures and took VIPs out of service only during the actual fail over and fail back.

Test Procedure

The procedure used to perform the DC NetApp Cluster Failover and Failback test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Perform the component failure by rebooting the primary filer.

Step 5 Verify ACE and CSM report failed probes and whether they brought the VIP out of service.

Step 6 Verify probes succeed and new connections are accepted for application hosts on ACE and CSM.

Step 7 Once the primary filer reboots, do a cluster giveback on the partner filer.

Step 8 Verify ACE and CSM report failed probes and whether they brought the VIP out of service.

Step 9 Verify after the giveback that ACE and CSM are again forwarding traffic to application hosts.

Step 10 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect all existing application traffic to be disrupted during the failover and resume after failover automatically.
- We expect some existing application traffic to be disrupted during the failback and resume after failback automatically.
- We expect ACE and CSM to recognize the filer recovery and bring it back in service after both failover and failback.

Results

DC NetApp Cluster Failover and Failback passed.

DC Primary Exchange Host Power Failure and Recovery

This test verified the functionality of Exchange during a physical application host failure. This involved sending DCAP application traffic from each branch to both Data Centers, power cycling an Exchange primary cluster host in one data center, and capturing the key metrics such as: impact to the application, lost transactions, and measured throughput.

For this test 15 minutes of application traffic was run. The traffic is optimized by WAAS but does not go through a load balancer. The primary Exchange cluster node was power cycled and it was verified that the existing connections to the application host were terminated and after failover the other cluster node serviced all new connections. After the power cycled host recovered, a manual failback to the server was performed and it was verified that the host accepted new connections./p>

Test Procedure

The procedure used to perform the DC Primary Exchange Host Power Failure and Recovery test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Perform the component failure by powering down the primary Exchange cluster host at least 2 minutes.

Step 5 Verify all new connections are redirected to the other cluster node.

Step 6 Power on the primary cluster host.

Step 7 When the host comes back online, fail back Exchange using Cluster Administrator.

Step 8 Verify the primary cluster host is servicing all new connections.

Step 9 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing application traffic connected to the physical host to fail.
- We expect new connections to be serviced by the other cluster node once failover has occurred.
- We expect the power cycled node to accept new connections once failback has occurred.

Results

DC Primary Exchange Host Power Failure and Recovery passed.

ESX Host Power Failure and Recovery ACE Redirection

This test verified the functionality of the application(s) during a Physical ESX host failures. This involved sending DCAP application traffic from each branch to both Data Centers, power cycling an application host and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and ACE load balances connections in a round robin fashion. One of the physical ESX blade server hosts is power cycled and it was verified that the existing connections to the application host were terminated and after a probe failure by ACE. Due to this probe failure, the ACE takes the host out of service. All the new connections will be sent to the other application hosts until the power cycled host recovers. When the host comes back online it was verified that ACE brings the physical host back in service and accepts new connections.

Test Procedure

The procedure used to perform the ESX Host Power Failure and Recovery ACE Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total user connections to the application host from ACE by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 5 Perform the component failure by powering down the ESX blade host via the HP Onboard Administrator thus simulating a crash.

Step 6 Verify the ACE has brought the host out of service after the component failure by issuing the **show probe ORACLE_WEB_PAGE_CHECK** and commands on the ACE.

Step 7 Verify all new connections are redirected to the other application hosts by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 8 Monitor the status of the ESX host and once it comes back online power up the application hosts from the Virtual Center Infrastructure client.

Step 9 When the host comes back online, start the application services by issuing the **./adstrtal.sh apps/apps** command.

Step 10 Verify that the ACE brings the host back inservice once the application services are up by issuing the **show probe ORACLE_WEB_PAGE_CHECK** command.

Step 11 Verify the ACE is again sending requests to the host by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 12 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing application traffic connected to the physical host to fail.
- We expect new connections to continue to pass after the component failure has occurred.
- We expect ACE to recognize the physical host recovery and bring it back in service to accept new connection.

Results

ESX Host Power Failure and Recovery ACE Redirection passed.

ESX Host Power Failure and Recovery CSM Redirection

This test verified the functionality of the application(s) during a physical ESX host failures. This involved sending DCAP application traffic from each branch to both Data Centers, power cycling an application host and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and CSM load balances connections in a round robin fashion. One of the ESX blade server hosts is power cycled and it was verified that the existing connections to the application host were terminated and after a probe failure by CSM. Due to this probe failure the CSM takes the host out of service. All the new connections will be sent to the other application hosts until the power cycled host recovers. When the ESX host comes back online it was verified that CSM brings the ESX virtual hosts back in service and accepts new connections./p>

Test Procedure

The procedure used to perform the ESX Host Power Failure and Recovery CSM Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total user connections to the application host from CSM by issuing the **show mod csm 2 serverfarms name oracle-all detail** command.

Step 5 Perform the component failure by powering down the application blade host via the HP Onboard Administrator for at least 2 minutes.

Step 6 Verify the CSM has brought the host out of service after the component failure by issuing the **show mod csm 2 probe name oracle detail** and commands on the CSM.

Step 7 Verify all new connections are redirected to the other application hosts by issuing the **show mod csm 2 serverfarms name oracle-all detai** command.

Step 8 Power on the application host in the HP Onboard Adminstrator GUI.

Step 9 When the host comes back online, start the application services by issuing the **./adstrtal.sh apps/apps** command.

Step 10 Verify that the CSM brings the host back inservice once the application services are up by issuing the **show mod csm 2 probe name oracle detail** command.

Step 11 Verify the CSM is again sending requests to the host by issuing the **show mod csm 2 serverfarms name oracle-all detai** command.

Step 12 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing application traffic connected to the virtual host(s) to fail.
- We expect new connections to continue to pass after the component failure has occurred.
- We expect the CSM to recognize the virtual host recovery and bring it back in service to accept new connections.

Results

ESX Host Power Failure and Recovery CSM Redirection passed.

Graceful Host Shutdown and Recovery ACE Redirection

This test verified the functionality of the application(s) during a virtual machine(VM) graceful application host shutdown and recovery. This involved sending DCAP application traffic from each branch to both Data Centers, gracefully removing one of the application hosts from the ACE serverfarm and re-introducing it after a reboot while capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and ACE load balances connections in a round robin fashion. A VM host was gracefully shutdown and it was verified the ACE stopped sending traffic to it once the probe failed.

Test Procedure

The procedure used to perform the Graceful Host Shutdown and Recovery ACE Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total user connections to VM application host from ACE by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 5 Perform the graceful removal of the application host on the ACE by issuing the **inservice standby** commands.

Step 6 Shutdown the application services on one of the application nodes.

Step 7 Verify the ACE has brought the host out of service by issuing the **show probe ORACLE_WEB_PAGE_CHECK** and commands on the ACE.

Step 8 Verify all new connections are redirected to the other application hosts by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 9 Reboot the application host by issuing the command **shutdown -r** .

Step 10 When the host comes back online, start the application services by issuing the **./adstrtal.sh apps/apps** command.

Step 11 Configure the ACE to bring the application host back inservice by issuing the **no inservice standby** command and the **inservice** command.

Step 12 Verify the ACE is again sending requests to the host by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 13 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect new connections to being to load balance to only the inservice application hosts.
- We expect after bringing the host back inservice it will begin recieving new connections.

Results

Graceful Host Shutdown and Recovery ACE Redirection passed.

Graceful Host Shutdown and Recovery CSM Redirection

This test verified the functionality of the application(s) during a Oracle virtual machine(VM) graceful application host shutdown and recovery. This involved sending DCAP application traffic from each branch to both Data Centers, gracefully removing one of the application hosts from the CSM serverfarm and re-introducing it after a reboot while capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and CSM load balances connections in a round robin fashion. An Oracle host was first manually taken out of service by the CSM. The Oracle application service was then stopped and the host rebooted. Once the host was back online the application was restarted and brought back in service manually by the CSM. It was verified that requests did not get sent to the host during the manual shutdown and recovery period until the host was healthy and back online.

Test Procedure

The procedure used to perform the Graceful Host Shutdown and Recovery CSM Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total number of user connections to the Oracle application host, dcap-dcb-oraapp01.dcap.com(201.1.33.16), from CSM into the service switch by issuing the **show mod csm 2 serverfarms name oracle-all detail** command. The VM host real server name is DOT16. The number of connections should be greater than one showing that the server is in fact being used.

Step 5 Perform the graceful removal of the real server DOT16 on the CSM by issuing the folowing commands:

```
configure terminal
module contentSwitchingModule 2
serverfarm ORACLE-ALL
real name DOT16
inservice standby
end
```

Step 6 On the application host that was just put into standby mode, log in via SSH and shutdown the Oracle application services gracefully by issuing the **./adstpall.sh apps/apps** command.

Step 7 Verify the CSM has brought the DOT16 real server out of service by issuing the **show mod csm 2 probe name oracle detail** command on the CSM.

Step 8 Verify all new connections are redirected to the other application hosts by issuing the **show mod csm 2 serverfarms name oracle-all detail** command. It should be expected however that no new connections are sent to the host that was brought out of service. Because the GSS is load balancing new connections to both data centers in a round robin fashion and the CSM is also load balancing incoming connections withing the Data Center it is indeterministic the number of connections to expect by the other hosts.

Step 9 Reboot the Oracle application host, dcap-dcb-oraapp01.dcap.com, by issuing the **shutdown -r now** command.

Step 10 When the host comes back online, start the application services by issuing the **./adstrtal.sh apps/apps** command.

Step 11 Configure the CSM to bring the application host back inservice by issuing the **no inservice standby** and **inservice** commands.

Step 12 Verify the CSM is again sending requests to the host by issuing the **show mod csm 2 serverfarms name oracle-all detail** command.

Step 13 Once the traffic has completed save and analyze the results to verify the expected end user experience. There should be no failed transactions.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect new connections to be load balanced to only the inservice application hosts.
- We expect after bringing the host back inservice it will begin receiving new connections.

Results

Graceful Host Shutdown and Recovery CSM Redirection passed.

Physical Host Power Failure and Recovery- ACE Redirection

This test verified the functionality of the Application(s) during a Physical application host failures. This involved sending DCAP application traffic from each branch to both Data Centers, power cycling an application host and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and ACE load balances connections in a round robin fashion. One of the physical blade server hosts is power cycled and it was verified that the existing connections to the application host were terminated and after a probe failure by ACE. Due to this probe failure the ACE takes the host out of service. All the new connections will be sent to the other application hosts until the power cycled host recovers. When host comes back online it was verified that ACE brings the physical host back in service and accepts new connections./p>

Test Procedure

The procedure used to perform the Physical Host Power Failure and Recovery- ACE Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total user connections to the application host from ACE by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 5 Perform the component failure by powering down the application blade host via the HP Onboard Administrator for at least 2 minutes.

Step 6 Verify the ACE has brought the host out of service after the component failure by issuing the **show probe ORACLE_WEB_PAGE_CHECK** and commands on the ACE.

Step 7 Verify all new connections are redirected to the other application hosts by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 8 Power on the application host in the HP Onboard Adminstrator GUI.

Step 9 When the host comes back online, start the application services by issuing the **./adstrtal.sh apps/apps** command.

Step 10 Verify that the ACE brings the host back inservice once the application services are up by issuing the **show probe ORACLE_WEB_PAGE_CHECK** command.

Step 11 Verify the ACE is again sending requests to the host by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 12 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing application traffic connected to the physical host to fail.
- We expect new connections to continue to pass after the component failure has occurred.
- We expect ACE to recognize the physical host recovery and bring it back in service to accept new connection.

Results

Physical Host Power Failure and Recovery- ACE Redirection passed.

VM Application Host Power Failure and Recovery ACE Redirection

This test verified the functionality of the Application(s) during a Virtual Machine(VM) Application host failures. This involved sending DCAP application traffic from each branch to both Data Centers, power cycling a VM application host and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and ACE load balances connections in a round robin fashion. One of the VM hosts is power cycled and it was verified that the existing connections to the application host were terminated and after a probe failure by ACE. Due to this probe failure the ACE takes the host out of service. All the new connections will be sent to the other application hosts until the power cycled host recovers. When host comes back online it was verified that ACE brings the VM host back in service and accepts new connections./p>

Test Procedure

The procedure used to perform the VM Application Host Power Failure and Recovery ACE Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total user connections to VM application host from ACE by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 5 Perform the component failure by powering down one application host in Virtual Center for at least 2 minutes.

Step 6 Verify the ACE has brought the host out of service after the component failure by issuing the **show probe ORACLE_WEB_PAGE_CHECK** and commands on the ACE.

Step 7 Verify all new connections are redirected to the other application hosts by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 8 Power on the application host in Virtual Center after 2 mins.

Step 9 When the host comes back online, start the application services

Step 10 Verify that the ACE brings the host back inservice once the application services are up by issuing the **show probe ORACLE_WEB_PAGE_CHECK** command.

Step 11 Verify the ACE is again sending requests to the host by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 12 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing application traffic connected to VM host to fail.
- We expect new connections to continue to pass after the component failure has occurred.
- We expect ACE to recognize the VM host recovery and bring it back in service to accept new connection.

Results

VM Application Host Power Failure and Recovery ACE Redirection passed.

VM Host Power Failure and Recovery CSM Redirection

This test verified the functionality of the application(s) during a virtual machine application host failures. This involved sending DCAP application traffic from each branch to both Data Centers, power cycling a virtual host and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and CSM load balances connections in a round robin fashion. One of the virutal hosts is power cycled via the Virtual Center Infrastructure client and it was verified that the existing connections to the application host were terminated and after a probe failure by CSM. Due to this probe failure the CSM takes the host out of service. All the new connections will be sent to the other application hosts until the power cycled host recovers. When host comes back online it was verified that CSM brings the virtual host back in service and accepts new connections.

Test Procedure

The procedure used to perform the VM Host Power Failure and Recovery CSM Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total user connections to the application host from CSM by issuing the **mod csm 2 serverfarms name oracle-all detail** command.

Step 5 Perform the component failure by powering down the application blade host via the Virtual Center Infrastructure client for at least 2 minutes.

Step 6 Verify the CSM has brought the host out of service after the component failure by issuing the **show mod csm 2 probe name oracle detail** and commands on the CSM.

Step 7 Verify all new connections are redirected to the other application hosts by issuing the **show mod csm 2 serverfarms name oracle-all detai** command.

Step 8 Power on the application host via the Virtual Infrasturcture client.

Step 9 When the host comes back online, start the application services by issuing the **./adstrtal.sh apps/apps** command.

Step 10 Verify that the CSM brings the host back inservice once the application services are up by issuing the **show mod csm 2 probe name oracle detail** command.

Step 11 Verify the CSM is again sending requests to the host by issuing the **show mod csm 2 serverfarms name oracle-all detai** command.

Step 12 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 13 Stop background scripts to collect final status of network devices and analyze for error.

Step 14 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing application traffic connected to the VM host to fail.

- We expect new connections to continue to pass after the component failure has occurred.
- We expect the CSM to recognize the VM host recovery and bring it back in service to accept new connections.

Results

VM Host Power Failure and Recovery CSM Redirection passed.

VMotion Host CSM Redirection

This test verified the functionality of the application(s) during a virtual machine(VM) VMotion event. This involved sending DCAP application traffic from each branch to both Data Centers, power cycling a VM application host and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and CSM load balances connections in a round robin fashion. One of the VM hosts is migrated with VMotion to a clustered ESX host and then migrated back. It was verified that the existing connections to the application host were not terminated and no probe failures were seen by CSM. All the new connections should continue to load balance to each application host.

Test Procedure

The procedure used to perform the VMotion Host CSM Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total user connections to the VM application host from the CSM by issuing the **show mod csm 2 serverfarms name oracle-all detail** command.

Step 5 Prepare for the scheduled maintenance by bringing down any virtual host on the ESX server that is not able to be migrated. Perform the application host migration by migrating from one ESX host to another by using the Virtual Center Infrastructure client.

Step 6 Verify the CSM has not brought the host out of service after the migration by issuing the **show mod csm 2 probe name oracle detail** and commands on the CSM. For this migration no probes should fail.

Step 7 Reboot the ESX host to simulate a maintenance event.

Step 8 After the ESX server reboots, migrate the application host back to its initial ESX host in the Virtual Center Infrastructure client. Also, restart any VM's that were shut down prior to the reboot.

Step 9 Verify that the CSM keeps the host inservice during the migration by issuing the **show mod csm 2 probe name oracle detail** command.

Step 10 Verify the CSM continues sending requests to the host by issuing the **show mod csm 2 serverfarms name oracle-all detail** command.

Step 11 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect no existing application traffic connected to the VM host to fail.
- We expect new connections to continue to pass after the VM has been migrated from one ESX host to the other.
- We expect CSM continue to load balance new connections to the migrated VM during and after the migration.

Results

VMotion Host CSM Redirection passed.

VMotion Host-ACE Redirection

This test verified the functionality of the application(s) during a virtual machine(VM) VMotion event. This involved sending DCAP application traffic from each branch to both Data Centers, power cycling a VM application host and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load balanced across 3 application hosts and ACE load balances connections in a round robin fashion. One of the VM hosts is migrated with VMotion to a clustered ESX host and then migrated back. It was verified that the existing connections to the application host were not terminated and no probe failures were seen by ACE. All the new connections should continue to load balance to each application host.

Test Procedure

The procedure used to perform the VMotion Host-ACE Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the total user connections to the VM application host from ACE by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 5 Prepare for the scheduled maintenance by bringing down any virtual host on the ESX server that is not able to be migrated. Perform the application host migration by migrating from one ESX host to another by using the Virtual Center Infrastructure client.

Step 6 Verify the ACE has not brought the host out of service after the migration by issuing the **show probe ORACLE_WEB_PAGE_CHECK** and commands on the ACE. For this migration no ACE probes should fail.

Step 7 Reboot the ESX host to simulate a maintenance event.

Step 8 After the ESX server reboots, migrate the application host back to its initial ESX host in the Virtual Center Infrastructure client. Also, restart any VM's that were shut down prior to the reboot.

Step 9 Verify that the ACE keeps the host inservice during the migration by issuing the **show probe ORACLE_WEB_PAGE_CHECK** command.

Step 10 Verify the ACE continues sending requests to the host by issuing the **show serverfarm ORAAPP_ORACLE_FARM** command.

Step 11 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect no existing application traffic connected to the VM host to fail.
- We expect new connections to continue to pass after the VM has been migrated from one ESX host to the other.
- We expect ACE continue to load balance new connections to the migrated VM during and after the migration.

Results

VMotion Host-ACE Redirection passed.

Baseline

This test provides a baseline for the network when it is in a steady state.

This section contains the following topics:

Generate Application Traffic for 15 Minutes

- This test verified the functionality of the applications during normal operations (that is, no error conditions in the network). This involved sending DCAP application traffic from each branch client for 15 minutes and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec). These metrics are used as baselines for other tests in which application traffic is generated in 15-minute increments.

Test Procedure

The procedure used to perform the Generate Application Traffic for 15 Minutes test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 5 Stop background scripts to collect final status of network devices and analyze for error.

Step 6 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass for the entire duration of the test.
- We expect no impact to the end user.

Results

Generate Application Traffic for 15 Minutes passed.

CSM Module

This suite of tests focused on high availability scenarios focusing on the CSM module.

This section contains the following topics:

Graceful CSM Module Failover

This test verified the functionality of the Application(s) during primary CSM FT switchover. This involved sending DCAP application traffic from each branch, forcing a CSM module FT switchover and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. On the CSM preempt was un-configured so that a graceful ft switchover could be done. It was verified that the standby CSM became active. Once the standby CSM became active we gracefully failed back to the initial active CSM.

Test Procedure

The procedure used to perform the Graceful CSM Module Failover test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 To perform a graceful ft failover on the CSM preempt must first be unconfigured by issuing the **no preempt** command on both the active and standby CSM.

Step 4 Initiate the test traffic for a duration of 10 minutes.

Step 5 Verify the state of the CSM modules by issuing the **show mod csm 2 ft** command.

Step 6 Perform the component failure by issuing the **clear mod csm 2 ft active** command.

Step 7 Verify the state of the new active CSM by issuing the **show mod csm 2 ft** command.

Step 8 Perform a graceful switchover to the initial active CSM by issuing the **clear mod csm 2 ft active** command on the new active CSM.

Step 9 Verify the initially active CSM once again is active by issuing the **show mod csm 2 ft** command. Also verify the initial standby CSM is again in the standby state.

Step 10 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 11 Reconfigure preempt on both the active and standby CSM by issuing the **preempt** command.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure has occurred.
- We expect minimal impact to the end user. The potential for short lived HTTP flows to fail during failover/failback.
- We expect the standby CSM to become primary during power cycle and revert to standby after original primary comes active.

Results

Graceful CSM Module Failover passed.

Power Cycle Primary CSM Module

This test verified the functionality of the Application(s) during primary CSM module power cycle. This involved sending DCAP application traffic from each branch, power cycling CSM module and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The primary CSM was powered off then back on and it was verified that the standby CSM became active. Once the power cycled CSM comes back online it is verified that it preempts and once again become the active CSM.

Test Procedure

The procedure used to perform the Power Cycle Primary CSM Module test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify the state of the CSM modules by issuing the **show mod csm 2 ft** command.

Step 5 Perform the component failure by issuing the **no power enable module *module*** command on dcb-ss-1. After 10 seconds power the module back on with the **power enable module *module*** command.

Step 6 Verify the state of the new active CSM by issuing the **show mod csm 2 ft** command.

Step 7 Verify the initially active CSM comes back online and preempts, once again becoming the active CSM by issuing the **show mod csm 2 ft** command. Also verify the initial standby CSM is again in the standby state.

Step 8 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure has occurred.
- We expect minimal impact to the end user. The potential for short lived HTTP flows to fail during failover/failback.
- We expect the standby CSM to become primary during power cycle and revert to standby after original primary comes active.

Results

Power Cycle Primary CSM Module passed.

Reset Primary CSM Module

This test verified the functionality of the Application(s) during primary CSM module failure. This involved sending DCAP application traffic from each branch, power cycling CSM module and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The primary CSM was power cycled and it was verified that the standby CSM became active. Once the power cycled CSM comes back online it should preempt and once again become the active CSM.

Test Procedure

The procedure used to perform the Reset Primary CSM Module test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify that preempt is enabled on both CSM modules by issuing the **show mod csm 2 ft** command.

Step 5 Verify the state of the CSM modules by issuing the **show mod csm 2 ft** command.

Step 6 Perform the component failure by issuing the **hw-module module *module*** reset command on dca-agg-1.

Step 7 Verify the state of the new active ACE by issuing the **show ft group summary** command.

Step 8 Verify the initially active CSM comes back online and preempts, once again becoming the active CSM by issuing the **show mod csm 2 ft** command. Also verify the initial standby CSM is again in the stanby state.

Step 9 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 10 Stop background scripts to collect final status of network devices and analyze for error.

Step 11 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the component failure has occurred.
- We expect minimal impact to the end user. The potential for short lived HTTP flows to fail during failover/failback.
- We expect the standby CSM to become primary during power cycle and revert to standby after original primary comes active.

Results

Reset Primary CSM Module passed.

Device Failure

This suite of tests focused on high availability scenarios focusing on device failure and recovery.

This section contains the following topics:

DCa Access 6k Reload

This test verified the functionality of the application(s) during a Catalyst 6500 access switch device failure. This involved sending DCAP application traffic from each branch and failing a core switch while measuring: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test application traffic was run for 15 minutes. One of the Catalyst 6500 switches in DCA actively passing traffic was failed and then recovered. Because NIC bonding was in place traffic was unaffected it was verified that minimal impact to the end user was seen as the traffic continued to pass through the a access switch.

Test Procedure

The procedure used to perform the DCa Access 6k Reload test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Reload one of the core switches actively passing traffic by issuing the **reload** command.

Step 5 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure has occurred.
- We expect minimal impact to the end user.

Results

DCa Access 6k Reload passed.

DCa Aggregation Switch Reload

This test verified the functionality of the application(s) during a aggregation switch failure. This involved sending DCAP application traffic from each branch failing the service switch housing the active service modules: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test application traffic was run for 15 minutes. The active HSRP aggregation switch in DCA was failed.

Test Procedure

The procedure used to perform the DCa Aggregation Switch Reload test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Reload the active HSRP aggregation switch by issuing the **reload** command.

Step 5 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the component failure has occurred.
- We expect minimal impact to the end user.

Results

DCa Aggregation Switch Reload passed.

DCa Core Reload

This test verified the functionality of the application(s) during a core device failure. This involved sending DCAP application traffic from each branch and failing a core switch while measuring: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test application traffic was run for 15 minutes. Once of the core switches in DCA actively passing traffic was failed and then recovered. It was verified that no impact to the end user was seen.

Test Procedure

The procedure used to perform the DCa Core Reload test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Reload one of the core switches actively passing traffic by issuing the **reload** command.

Step 5 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure has occurred.
- We expect minimal impact to the end user.

Results

DCa Core Reload passed.

DCb Access 4k Reload

This test verified the functionality of the application(s) during a Catalyst 4948 access switch device failure. This involved sending DCAP application traffic from each branch and failing a core switch while measuring: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test application traffic was run for 15 minutes. Once of the Catalyst 4948 switches in DCB actively passing traffic was failed and then recovered. Because NIC bonding was in place traffic was unaffected it was verified that minimal impact to the end user was seen as the traffic continued to pass through the a access switch.

Test Procedure

The procedure used to perform the DCb Access 4k Reload test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Reload one of the core switches actively passing traffic by issuing the **reload** command.

Step 5 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure has occurred.
- We expect minimal impact to the end user.

Results

DCb Access 4k Reload passed.

DCb Aggregation Switch Reload

This test verified the functionality of the application(s) during a aggregation switch failure. This involved sending DCAP application traffic from each branch failing the service switch housing the active service modules: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test application traffic was run for 15 minutes. The active HSRP aggregation switch in DCB was failed.

Test Procedure

The procedure used to perform the DCb Aggregation Switch Reload test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Reload the active HSRP aggregation switch by issuing the **reload** command.

Step 5 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 6 Stop background scripts to collect final status of network devices and analyze for error.

Step 7 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the component failure has occurred.
- We expect minimal impact to the end user.

Results

DCb Aggregation Switch Reload passed.

GSS Interface Shutdown DCa

This test verified the functionality of the application(s) during a GSS failure at DCA. This involved sending DCAP application traffic from each branch failing the service switch housing the active service modules: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test application traffic was run for 15 minutes. One of the two GSS's in DCA was taken offline.

Test Procedure

The procedure used to perform the GSS Interface Shutdown DCa test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Stop the GSS and shutdown the GSS interface on a GSS in DCA by issuing the **gss stop**and the **shutdown** command.

Step 5 Once the traffic has run for approximately 10 minutes since the GSS failure, bring the failed GSS back online by issuing the **no shutdown**and the **gss start** command.

Step 6 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the component failure has occurred.
- We expect minimal impact to the end user.

Results

GSS Interface Shutdown DCa passed.

GSS Interface Shutdown DCb

This test verified the functionality of the application(s) during a aggregation switch failure. This involved sending DCAP application traffic from each branch failing the service switch housing the active service modules: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test application traffic was run for 15 minutes.

Test Procedure

The procedure used to perform the GSS Interface Shutdown DCb test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Stop the GSS and shutdown the GSS interface on a GSS in DCB by issuing the **gss stop**and the **shutdown** command.

Step 5 Once the traffic has run for approximately 10 minutes since the GSS failure, bring the failed GSS back online by issuing the **no shutdown**and the **gss start** command.

Step 6 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 7 Stop background scripts to collect final status of network devices and analyze for error.

Step 8 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the component failure has occurred.
- We expect minimal impact to the end user.

Results

GSS Interface Shutdown DCb passed.

Link Failure

This suite of tests focused on high availability scenarios focusing on link failure and recovery.

This section contains the following topics:

- DCa Link Failures, page 17-56
- DCb Link Failures, page 17-57

DCa Link Failures

This test verified the functionality of the application(s) during various active redundant link failures. This involved sending DCAP application traffic from each branch and shutting down and bringing back up links that have a redundant link in place while measuring: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test application traffic was run for 15 minutes. Each link in place for redundancy was brought down causing the redundant data path to take over, then the downed link was brought back up. It was verified that the application end user impact was minimal.

Test Procedure

The procedure used to perform the DCa Link Failures test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Shut down the link between the two core switches by issuing the **shutdown** command.

Step 5 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 6 Shut down the link between core-1 and agg-1 by issuing the **shutdown** command.

Step 7 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 8 Shut down the link between core-2 and agg-1 by issuing the **shutdown** command.

Step 9 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 10 Shut down one link of the port-channel between agg-1 and agg-2 by issuing the **shutdown** command.

Step 11 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 12 Shut down the other link of the port-channel between agg-1 and agg-2 by issuing the **shutdown** command.

Step 13 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 14 Shut down the link between agg-1 and acc-6k-2 by issuing the **shutdown** command.

Step 15 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 16 Shut down the link between agg-1 and acc-4k-1 by issuing the **shutdown** command.

Step 17 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 18 Shut down the link between agg-2 and acc-4k-2 by issuing the **shutdown** command.

Step 19 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 20 Shut down one port of the port-channel between agg-1 and the dca-hp-switch-2 by issuing the **shutdown** command.

Step 21 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 22 Shut down a fiber channel link belonging to a SAN port channel between a core switch and all edge switches by issuing the **shutdown** command.

Step 23 Bring back up the port channel links that were previously shutdown by issuing the **no shutdown** command.

Step 24 Shut down a fiber channel link belonging to a port channel between two transit switches by issuing the **shutdown** command.

Step 25 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 26 Shut down an FCIP link belonging to a port channel between two transit switches by issuing the **shutdown** command.

Step 27 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 28 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 29 Stop background scripts to collect final status of network devices and analyze for error.

Step 30 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure has occurred.
- We expect minimal impact to the end user.

Results

DCa Link Failures passed.

DCb Link Failures

This test verified the functionality of the application(s) during various active redundant link failures. This involved sending DCAP application traffic from each branch and shutting down and bringing back up links that have a redundant link in place while measuring: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test application traffic was run for 15 minutes. Each link in place for redundancy was brought down causing the redundant data path to take over, then the downed link was brought back up. It was verified that the application end user impact was minimal.

Test Procedure

The procedure used to perform the DCb Link Failures test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Shut down one link of the Portchannel between agg-1 and ss-1 by issuing the **shutdown** command.

Step 5 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 6 Shut down the other link of the Portchannel between agg-1 and ss-1 by issuing the **shutdown** command.

Step 7 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 8 Shut down one link of the port-channel between agg-1 and agg-2 by issuing the **shutdown** command.

Step 9 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 10 Shut down the other link of the port-channel between agg-1 and agg-2 by issuing the **shutdown** command.

Step 11 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 12 Shut down the link between agg-1 and acc-6k-2 by issuing the **shutdown** command.

Step 13 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 14 Shut down the link between agg-1 and acc-4k-1 by issuing the **shutdown** command.

Step 15 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 16 Shut down the link between agg-2 and acc-4k-2 by issuing the **shutdown** command.

Step 17 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 18 Shut down a fiber channel link belonging to a SAN port channel between a core switch and all edge switches by issuing the **shutdown** command.

Step 19 Bring back up the port channel links that were previously shutdown by issuing the **no shutdown** command.

Step 20 Shut down a fiber channel link belonging to a port channel between two transit switches by issuing the **shutdown** command.

Step 21 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 22 Shut down an FCIP link belonging to a port channel between two transit switches by issuing the **shutdown** command.

Step 23 Bring back up the link that was previously shutdown by issuing the **no shutdown** command.

Step 24 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 25 Reconfigure preempt on both the active and standby ACE by issuing the **preempt** command.

Step 26 Stop background scripts to collect final status of network devices and analyze for error.

Step 27 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure has occurred.
- We expect minimal impact to the end user.

Results

DCb Link Failures passed.

SAN

This suite of tests focused on high availability scenarios focusing on SAN device failure and recovery.

This section contains the following topics:

HP c-Class BladeSystem MDS 9124e Reload and Recovery

This test verified the functionality of application during a reset of a redundant HP BladeSystem MDS 9124e integrated SAN switch. This involved sending DCAP application traffic from each branch to both Data Centers, resetting an MDS 9124e in the HP BladeCenter in one data center, and capturing the key metrics such as: impact to the application, lost transactions, and measured throughput.

For this test 15 minutes of application traffic was run. The MDS 9124e was reloaded and it was verified that the application hosts were able to access SAN storage over redundant links without any disruption to the applications. After the reloaded switch recovered, it was verified that application traffic continued flowing uninterrupted and the paths through the switch were reestablished./p>

Test Procedure

The procedure used to perform the HP c-Class BladeSystem MDS 9124e Reload and Recovery test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Reload one of the MDS 9124e SAN switches.

Step 5 Verify applications are not impacted by the switch reload.

Step 6 When the switch comes back online, verify paths through the switch come back online.

Step 7 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing SAN-based applications to continue to access SAN storage over redundant paths while the switch is reloading.
- We expect paths through the reloaded switch to be reestablished once the switch is back up.

Results

HP c-Class BladeSystem MDS 9124e Reload and Recovery passed.

MDS Core Switch Reload and Recovery

This test verified the functionality of application during a reset of a redundant MDS 9513 SAN switch deployed as a core device. This involved sending DCAP application traffic from each branch to both Data Centers, resetting a core MDS 9513 in one data center, and capturing the key metrics such as: impact to the application, lost transactions, and measured throughput.

For this test 15 minutes of application traffic was run. The MDS was reloaded and it was verified that the application hosts were able to access redundant paths to SAN storage without any disruption to application services. After the reloaded switch recovered, it was verified that application traffic continued flowing uninterrupted and the paths through the reloaded switch were reestablished.

Test Procedure

The procedure used to perform the MDS Core Switch Reload and Recovery test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Reload one of the MDS core SAN switches.

Step 5 Verify applications are not impacted by the switch reload.

Step 6 When the switch comes back online, verify paths through the switch come back online.

Step 7 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 8 Stop background scripts to collect final status of network devices and analyze for error.

Step 9 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing SAN-based applications to continue to access SAN storage over redundant paths without disruption while the switch is reloading.

- We expect paths through the reloaded switch to be reestablished once the switch is back up.

Results

MDS Core Switch Reload and Recovery passed.

MDS Transit Switch Reload and Recovery

This test verified the functionality of application during a reset of a redundant MDS 9513 SAN switch deployed as a transit device (replication between data centers). This involved sending DCAP application traffic from each branch to both Data Centers, resetting a transit MDS 9513 in one data center, and capturing the key metrics such as: impact to the application, lost transactions, and measured throughput.

For this test 15 minutes of application traffic was run. The MDS was reloaded and it was verified that the application dhosts were able to access SAN storage without any disruption to application services and data replication continued through redundant paths in the transit fabrics. (NOTE: NetApp SnapMirror traffic only has a single path, so replication is deferred duing the switch outage.) After the reloaded switch recovered, it was verified that application and replication traffic continued flowing uninterrupted and the paths through the reloaded switch were reestablished.

Test Procedure

The procedure used to perform the MDS Transit Switch Reload and Recovery test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the application test traffic from each branch and SAN replication between the data centers are configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Reload one of the MDS core SAN switches.

Step 5 Verify applications are not impacted by the switch reload.

Step 6 Verify replication continues through redundant paths where applicable. EXCEPTION: NetApp SnapMirror traffic only has a single path, so replication is deferred duing the switch outage.

Step 7 When the switch comes back online, verify replication paths through the switch come back online.

Step 8 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect existing SAN-based replication to continue over redundant paths in the transit fabric without disruption while the switch is reloading. (EXCEPTION: NetApp SnapMirror traffic only has a single path, so replication is deferred duing the switch outage.)
- We expect replication paths through the reloaded switch to be reestablished once the switch is back up.

Results

MDS Transit Switch Reload and Recovery passed.

WAAS

This suite of tests focused on high availability scenarios focusing on WAE link and device failure and recovery.

This section contains the following topics:

WAE Link Failure and Recovery WCCPv2 Redirection

This test verified the functionality of the Application(s) during DCB WAE failures. This involved sending DCAP application traffic from each branch to both Data Centers, load balancing with ACE to 2 Core WAEs, power cycling one WAE module and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load-balanced to 2, WAE-7326's via the WCCPv2 with L2-redirect and Mask Assign at the WAN edge 6500. The WAE link that test traffic was being load balanced across was brought down and back up. It was verified that after the WCCP failure was detected traffic was once again optimized by the remaining WAE.

Test Procedure

The procedure used to perform the WAE Link Failure and Recovery WCCPv2 Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify which WAE WCCPv2 is load-balancing the traffic to by issuing the **show tfo connection summary** command on each in service WAE.

Step 5 Perform the component failure by issuing the **shutdown** command on the in-band interface for the WAE that is optimizing the application traffic.

Step 6 Verify on the WAN router that WCCPv2 takes the WAE out of service after it fails to receive 3 hello packets from the failed WAE by issuing the **show ip wccp 61 detail** and **show ip wccp 62 detail** commands.

Step 7 Verify the remaining in service WAE begins to optimize the application traffic with the **show tfo connection summary** command.

Step 8 Perform the component recovery by issuing the **no shutdown** command on the WAE interface that was brought down in the previous step.

Step 9 Verify WCCPv2 reconverges on the WAN router by issuing the **show ip wccp 61 detail** and **show ip wccp 62 detail** commands.

Step 10 Verify all application traffic is once again load balanced and optimized across one or more WAE's by issuing the **show tfo connection summary** command.

Step 11 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 12 Stop background scripts to collect final status of network devices and analyze for error.

Step 13 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure/recovery has occurred.
- We expect 30 seconds of traffic loss to occur when the WAE is brought out of service.
- We expect 30 seconds of traffic loss to occur when the WAE comes back online.
- We expect traffic to once again become optimized when the WAE comes back online.

Results

WAE Link Failure and Recovery WCCPv2 Redirection passed.

WAE Power Failure and Recovery ACE Redirection

This test verified the functionality of the DC Application(s) during an ACE directed WAE failure and recovery. This involved sending DCAP application traffic from each branch to both Data Centers, load balanced in DCA with ACE to two Core WAEs, power cycling one WAE module and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load-balanced to 2, WAE-7326's with the ACE module in the DCA aggregation layer. One of the in service WAEs was power cycled and it was verified first that after the 3 second probe failure ACE stopped passing traffic to the failed WAE, passing only to the remaining core WAE. When the failed WAE comes back online it was verified that traffic once again load balanced traffic to it and the remaining in-service WAE, and that connections were optimized to both.

Test Procedure

The procedure used to perform the WAE Power Failure and Recovery ACE Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify ACE is load-balancing application traffic from each branch client across both core WAEs and is that the traffic is optimized by issuing the **show tfo connection summary** command.

Step 5 Perform the component failure by issuing the **reload** command on the WAE. Issue the **ctrl+c** command so that connections are not gracefully exited thus simulating a true failure.

Step 6 Verify the ACE has brought the WAE out of service after the 3s probe failure and that it is now redirecting traffic to the remaining WAE only by issuing the **show probe WAE_ICMP** command on the ACE and the **show tfo connection summary** command on the WAE.

Step 7 Verify all new connections are optimized by the remaining in-service WAE by issuing the **show tfo connection summary** command.

Step 8 As the WAE comes back online verify the ACE brings it back in service by issuing the **show probe WAE_ICMP** command.

Step 9 Verify all application traffic is once again load balanced and optimized across each WAE by issuing the **show tfo connection summary** command.

Step 10 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the component failure/recovery has occurred.
- We expect 3 seconds of traffic loss to occur when each WAE is brought out of service. This is the amount of time it takes ACE to timeout.
- We expect ACE to recognize the WAE failure and bring it out of service.
- We expect ACE to recognize the WAE recovery and bring it back in service.

Results

WAE Power Failure and Recovery ACE Redirection failed. The following failures were noted: CSCsl68531.

WAE Power Failure and Recovery WCCPv2 Redirection

This test verified the functionality of the Application(s) during DCB WAE failures. This involved sending DCAP application traffic from each branch to both Data Centers, load balancing with ACE to 2 Core WAEs, power cycling one WAE module and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load-balanced to 2, WAE-7326's via the WCCPv2 with L2-redirect and Mask Assign at the WAN edge 6500. The WAE that test traffic was being load balanced to was power cycled and it was verified that after the WCCP failure was detected traffic was once again optimized by the remaining WAE.

Test Procedure

The procedure used to perform the WAE Power Failure and Recovery WCCPv2 Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify which WAE WCCPv2 is load-balancing the traffic to by issuing the **show tfo connection summary** command on each in service WAE. .

Step 5 Perform the component failure by issuing the **reload** command on the WAE. Issue the **ctrl+c** command so that connections are not gracefully exited thus simulating a true failure.

Step 6 Verify on the WAN router that WCCP takes the WAE out of service after it fails to receive 3 hello packets from the failed WAE by issuing the **show ip wccp 61 detail** and **show ip wccp 62 detail** commands.

Step 7 Verify the remaining in service WAE begins to optimize the application traffic with the **show tfo connection summary** command.

Step 8 When the power cycled WAE comes back online verify that WCCPv2 reconverges on the WAN router by issuing the **show ip wccp 61 detail** and **show ip wccp 62** detail commands.

Step 9 Verify all application traffic is once again load balanced and optimized across each WAE by issuing the **show tfo connection summary** command.

Step 10 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 11 Stop background scripts to collect final status of network devices and analyze for error.

Step 12 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure/recovery has occurred.
- We expect 30 seconds of traffic loss to occur when the WAE is brought out of service.
- We expect 30 seconds of traffic loss to occur when the WAE comes back online.
- We expect traffic to once again become optimized when the WAE comes back online.

Results

WAE Power Failure and Recovery WCCPv2 Redirection passed.

WAE Standby Interface Failure and Recovery ACE Redirection

This test verified the functionality of the Application(s) during a DCA WAE interface failure and recovery. This involved sending DCAP application traffic from each branch to both Data Centers, load balancing with ACE to 2 Core WAEs, failing one WAE active interface of the standby group and capturing the key metrics such as: impact to the applications, lost transactions, transaction response times, and measured throughput (txns/sec).

For this test 15 minutes of application traffic was run. The traffic is load-balanced to 2, WAE-7326's via the ACE appliance in the DCA aggregation. One of the in service WAEs active interface of the standby group was failed and it was verified that the WAE stayed online, no connections were lost, and that traffic continued to be load balanced and optimized to both core WAEs. The failed interface was then recovered and it was again verified that traffic continued, optimized, with no loss.

Test Procedure

The procedure used to perform the WAE Standby Interface Failure and Recovery ACE Redirection test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Verify that the Application test traffic from each branch is configured.

Step 3 Initiate the test traffic for a duration of 15 minutes.

Step 4 Verify ACE is load-balancing application traffic from each branch client across both core WAEs and is that the traffic is optimzed by issuing the **show tfo connection summary** command.

Step 5 Verify on the core WAE that the connection to active aggregation switch, GigabitEthernet1/0, is the standby group active interface by issuing the **show standby** command.

Step 6 Perform the component failure by issuing the **shutdown** command on the active aggregation switch interface connecting to the active interface on the WAE.

Step 7 Verify the initial standby interface on the core WAE has become active by issuing the **show standby** command.

Step 8 Verify all application traffic continues to be load balanced and optimized across each WAE by issuing the **show tfo connection summary** command.

Step 9 Perform the component recovery by issuing the **no shutdown** command on the active aggregation switch interface connecting to the active interface on the WAE.

Step 10 Verify the initial active interface on the core WAE stays in this standby state by issuing the **show standby** command.

Step 11 Verify all application traffic continues to be load balanced and optimized across each WAE by issuing the **show tfo connection summary** command.

Step 12 Fail the standby link on dca-agg-2 by issuing the **shutdown** command to bring the network back to the pre-test baseline.

Step 13 Once the traffic has completed save and analyze the results to verify the expected end user experience.

Step 14 Stop background scripts to collect final status of network devices and analyze for error.

Step 15 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect application traffic to continue to pass after the compenent failure/recovery has occurred.
- We expect no traffic loss to occur.

Results

WAE Standby Interface Failure and Recovery ACE Redirection passed.

Replication

The replication category comprises tests of various IP-based data center to data center replication mechanisms using Cisco optimizations such as WAAS. These mechanisms are not specific to a particular application.

This section contains the following topics:

NetApp

The NetApp replication tests include SnapMirror in both synchronous and asynchronous modes over IP with and without optimization by WAAS.

This section contains the following topics:

DC NetApp SnapMirror Async Disaster Recovery Replication over IP with WAAS

This test verified that Network Appliance (NetApp) Asynchronous SnapMirror over IP with WAAS worked as expected when replicating data for disaster recovery purposes. Prior to the replication test, NetApp asynchronous SnapMirror was configured to replicate from a filer in each data center to a filer in the other data center. In one data center, a Linux host accessed the filer via a fiber channel SAN LUN and in the other data center, a Linux host accessed the filer via NFS. Cisco Catalyst and MDS switches provided IP and fiber channel SAN connectivity. The data consisted of Microsoft Windows 2003 Server and Red Hat Enterprise Linux 4 operation system files and Oracle and Microsoft Exchange data files. A simulated 2500 km distance separated the source and target filers. The key metric gathered during testing was throughput. Also, after replication a failover was performed and the source and target files were compared to ensure data integrity.

Test Procedure

The procedure used to perform the DC NetApp SnapMirror Async Disaster Recovery Replication over IP with WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Clean up from previous test as needed.

Step 3 Initialize snapmirror relationships. NOTE: This step is only needed if a previous snapshot relationship does not exist.

Step 4 Check for WAAS acceleration and clear counters.

Step 5 Check latency and bandwidth settings on WAN emulator and set latency to 2500km.

Step 6 Start timer and copy test data.

Step 7 Stop time when copy and replication are complete and compute throughput. Also gather WAAS statistics.

Step 8 Fail over storage and verify integrity of the replication.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that NetApp SnapMirror asynchronous replication and fail over will be supported fully by the Catalyst and MDS replication switches using 100 km of latency.
- We expect that WAAS will accelerate SnapMirror traffic.
- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all application data will be replicated in support of a Recovery Point Objective (RPO) of 0 (due to allowing the asynchronous replication to complete before failing over).
- We expect no CPU or memory problems.

Results

DC NetApp SnapMirror Async Disaster Recovery Replication over IP with WAAS failed. The following failures were noted: CSCsh72271 and CSCsg79439.

DC NetApp SnapMirror Async Disaster Recovery Replication over IP without WAAS

This test verified that Network Appliance (NetApp) Asynchronous SnapMirror over IP without WAAS worked as expected when replicating data for disaster recovery purposes. Prior to the replication test, NetApp asynchronous SnapMirror was configured to replicate from a filer in each data center to a filer in the other data center. In one data center, a Linux host accessed the filer via a fiber channel SAN LUN and in the other data center, a Linux host accessed the filer via NFS. Cisco Catalyst and MDS switches provided IP and fiber channel SAN connectivity. The data consisted of Microsoft Windows 2003 Server and Red Hat Enterprise Linux 4 operation system files and Oracle and Microsoft Exchange data files. A simulated 2500 km distance separated the source and target filers. The key metric gathered during testing was throughput. Also, after replication a failover was performed and the source and target files were compared to ensure data integrity.

Test Procedure

The procedure used to perform the DC NetApp SnapMirror Async Disaster Recovery Replication over IP without WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Clean up from previous test as needed.

Step 3 Initialize snapmirror relationships. NOTE: This step is only needed if a previous snapshot relationship does not exist.

Step 4 Check that WAAS is not in the path.

Step 5 Check latency and bandwidth settings on WAN emulator and set latency to 2500km.

Step 6 Start timer and copy test data.

Step 7 Stop time when copy and replication are complete and compute throughput. Also gather WAAS statistics.

Step 8 Fail over storage and verify integrity of the replication.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that NetApp SnapMirror asynchronous replication and fail over will be supported fully by the Catalyst and MDS replication switches using 2500 km of latency.
- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all application data will be replicated in support of a Recovery Point Objective (RPO) of 0 (due to allowing the asynchronous replication to complete before failing over).
- We expect no CPU or memory problems.

Results

DC NetApp SnapMirror Async Disaster Recovery Replication over IP without WAAS passed.

DC NetApp SnapMirror Sync Disaster Recovery Replication over IP with WAAS

This test verified that Network Appliance (NetApp) Synchronous SnapMirror over IP with WAAS worked as expected when replicating data for disaster recovery purposes. Prior to the replication test, NetApp synchronous SnapMirror was configured to replicate from a filer in each data center to a filer in the other data center. In one data center, a Linux host accessed the filer via a fiber channel SAN LUN and in the other data center, a Linux host accessed the filer via NFS. Cisco Catalyst and MDS switches provided IP and fiber channel SAN connectivity. The data consisted of Microsoft Windows 2003 Server and Red Hat Enterprise Linux 4 operation system files and Oracle and Microsoft Exchange data files. A simulated 100 km distance separated the source and target filers. The key metric gathered during testing was throughput. Also, after replication a failover was performed and the source and target files were compared to ensure data integrity.

Test Procedure

The procedure used to perform the DC NetApp SnapMirror Sync Disaster Recovery Replication over IP with WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Clean up from previous test as needed.

Step 3 Initialize snapmirror relationships. NOTE: This step is only needed if a previous snapshot relationship does not exist.

Step 4 Check for WAAS acceleration and clear counters.

Step 5 Check latency and bandwidth settings on WAN emulator and set latency to 100km.

Step 6 Start timer and copy test data.

Step 7 Stop time when copy and replication are complete and compute throughput. Also gather WAAS statistics.

Step 8 Fail over storage and verify integrity of the replication.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that NetApp SnapMirror synchronous replication and fail over will be supported fully by the Catalyst and MDS replication switches using 100 km of latency.
- We expect that WAAS will accelerate SnapMirror traffic.
- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all application data will be replicated in support of a Recovery Point Objective (RPO) of 0 (due to synchronous replication).
- We expect no CPU or memory problems.

Results

DC NetApp SnapMirror Sync Disaster Recovery Replication over IP with WAAS failed. The following failures were noted: CSCsh72271 and CSCsg79439.

DC NetApp SnapMirror Sync Disaster Recovery Replication over IP without WAAS

This test verified that Network Appliance (NetApp) Synchronous SnapMirror over IP without WAAS worked as expected when replicating data for disaster recovery purposes. Prior to the replication test, NetApp synchronous SnapMirror was configured to replicate from a filer in each data center to a filer in the other data center. In one data center, a Linux host accessed the filer via a fiber channel SAN LUN and in the other data center, a Linux host accessed the filer via NFS. Cisco Catalyst and MDS switches provided IP and fiber channel SAN connectivity. The data consisted of Microsoft Windows 2003 Server and Red Hat Enterprise Linux 4 operation system files and Oracle and Microsoft Exchange data files. A

simulated 100 km distance separated the source and target filers. The key metric gathered during testing was throughput. Also, after replication a failover was performed and the source and target files were compared to ensure data integrity.

Test Procedure

The procedure used to perform the DC NetApp SnapMirror Sync Disaster Recovery Replication over IP without WAAS test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPU utilization of those network devices.

Step 2 Clean up from previous test as needed.

Step 3 Initialize snapmirror relationships. NOTE: This step is only needed if a previous snapshot relationship does not exist.

Step 4 Check that WAAS is not in the path.

Step 5 Check latency and bandwidth settings on WAN emulator and set latency to 100km.

Step 6 Start timer and copy test data.

Step 7 Stop time when copy and replication are complete and compute throughput. Also gather WAAS statistics.

Step 8 Fail over storage and verify integrity of the replication.

Step 9 Stop background scripts to collect final status of network devices and analyze for error.

Step 10 Verify that memory and the CPU did not suffer severe, sustained, or unacceptable impact.

Expected Results

The following test results are anticipated:

- We expect that NetApp SnapMirror synchronous replication and fail over will be supported fully by the Catalyst and MDS replication switches using 100 km of latency.
- We expect that all applications will have a Recovery Time Objective (RTO) of less than one hour.
- We expect that all application data will be replicated in support of a Recovery Point Objective (RPO) of 0 (due to synchronous replication).
- We expect no CPU or memory problems.

Results

DC NetApp SnapMirror Sync Disaster Recovery Replication over IP without WAAS passed.

APPENDIX **A**

SAN Implementation

This section contains detailed configuration information specific to each storage vendor represented in the DCAP SAN topology, including a description of the multipath and replication mechanisms and basic host and storage array configuration information.

EMC

This section has general and detailed information about the EMC DMX3 and CX3 frames used in the DCAP SAN topology. Following a brief general summary of results in Table A-1, Table A-2 has software and firmware information, Table A-3 has hardware information, and at the end is representative host device information showing redundant host paths and replicated devices.

General Summary

No issues were found in host connectivity or synchronous replication over FC, but asynchronous replication over FCIP with the DMX frames had some issues due to frame cache limitations. Some asynchronous replication results (all tests at 5000 km and the 100 km test with FCWA, compression, and encryption enabled) are affected by the logical links becoming suspended due to running out of DMX cache resources. This is caused by the high I/O rate and the limited cache, not by any MDS issues. The following configuration change (applied using the symconfigure CLI utility) deferred the suspensions at the cost of less throughput, but they still happened in the above mentioned tests:

set symmetrix rdfa_host_throttle_time = 30;

With the above setting, the amount of cache available for SRDF/A goes from the default (94% of 10.5 GB on the particular frames used) to 100% of 10.5 GB, and host I/O is paused for 30 seconds whenever cache reaches 99% full. EMC guidelines recommend against this setting, since applications can crash, but iorate/iometer in conjunction with PowerPath seemed to tolerate this. (Note: 10 seconds was too low to make much difference.)

The load balancing policy for port channels was SID/DID/OXID.

For load balancing and high availability, synchronous replication used two FC paths, each with a theoretical bandwidth of 4 Gbps, and asynchronous replication used four FCIP paths, each with a theoretical bandwidth of 311 Mbps (half an OC-12). For each I/O test, a Linux host and a Windows host were each performing 8 km writes in 2 separate processor threads as fast as possible in each data center to stress the replication topology.

Table A-1 summarizes the iometer and iorate results from representative base connectivity and synchronous and asynchronous replication tests.

Note These results are only valid to compare results with various MDS optimizations applied; they are not a reflection on the performance capabilities or quality of the storage arrays themselves.

Table A-1 EMC DMX3 Iometer/Iorate Results (average per host type)

Traffic Type	Distance	I/O type	Host	I/O per sec	MB per sec
Synchronous replication without FC write acceleration	0 km	8 KB sequential writes	Linux	2699	21.1
			Windows	2617	20.5
	100 km		Linux	1212	9.5
			Windows	1184	9.2
Synchronous replication with FC write acceleration	100 km		Linux	1820	14.3
			Windows	1792	14.0
Asynchronous replication alone	0 km		Linux	8521	66.6
			Windows	9231	72.1
	100 km		Linux	7528	58.8
			Windows	8522	66.6
	5000 km		*Linux	7351	57.4
			Windows	8667	67.7
Asynchronous replication with FCIP write acceleration	0 km		Linux	8472	66.1
			Windows	9207	71.8
	100 km		Linux	8290	64.8
			Windows	8998	70.3
	5000 km		*Linux	6410	50.1
			*Windows	8362	65.6
Asynchronous replication with FCIP compression	0 km		Linux	8275	64.6
			Windows	9187	71.8
	100 km		Linux	8127	63.5
			Windows	8822	68.9
	5000 km		*Linux	6510	50.8
			*Windows	8096	48.4
Asynchronous replication with FCIP encryption	0 km		Linux	8460	66.1
			Windows	8846	69.1
	100 km		Linux	7808	61.0
			Windows	8316	64.9
	5000 km		*Linux	5791	32.6
			*Windows	8608	66.3
Asynchronous replication with FCIP write acceleration, compression, and encryption	0 km		Linux	7152	55.9
			Windows	8270	64.6
	100 km		*Linux	9311	72.8
			*Windows	9644	75.3
	5000 km		*Linux	6618	51.7
			*Windows	8039	62.8

* Throughput affected by insufficient cache.

Table A-2 summarizes the EMC software and firmware configuration information.

Table A-2 EMC Software/Firmware Information

Platform	Software Component	Function	Location	Version
DMX	Symmetrix CLI (SYMCLI)	configuration and monitoring, replication control	Linux hosts (x86_64)	V6.4.2.03 (Edit Level: 845)
DMX	Symmetrix CLI (SYMCLI)	configuration and monitoring, replication control	Windows hosts (i386)	V6.4.2.3 (Edit Level: 845)
DMX	Enginuity Microcode	operating system	frame	5771 (168B0000), Patch date 06.14.2007, Patch level 99
CX	FLARE Code	operation system	frame	FLARE 24, Patch 11
DMX, CX	PowerPath	multipath	Linux hosts (x86_64)	5.0.0 (build 157)
DMX, CX	PowerPath	multipath	Windows hosts (i386)	5.1.0 (build 51)

Table A-3 summarizes the EMC hardware configuration information.

Table A-3 EMC Hardware Information

Platform	Hardware Component	Quantity	Comments
DMX	Frame	2	Serials 000190300320, 000190300321
	Cache	16384 MB	per frame
	Disk	60 @ 146 GB	per frame, FC
	Disk director (DA)	2	per frame
	Fiber director (FA)	2 @ 8 ports (2 Gbps)	per frame
CX	Frame	1	Serials CK200072700312
	Cache	2048 MB	per frame
	Disk	6 @ 300 GB	per frame, FC
	Storage Processors	2	per frame
	Fiber ports	4 (2 FC per SP @ 2Gbps	per frame

Network Appliance

This appendix has general and detailed information about the Network Appliance FAS6070 frames used in the DCAP SAN topology. Following a brief general summary of results in Table A-4, Table A-5 has software and firmware information, Table A-6 has hardware information, and at the end is representative host device information showing redundant host paths and replicated devices

General Summary

No issues were found in host connectivity, but three limitations in NetApp's SAN design and an operational issue significantly affected replication. The three design limitations are as follows:

1. NetApp's IPFC and FC-VI-based implementation of SAN-based replication does not allow any benefit from write acceleration over either fiber channel or FCIP.
2. NetApp's high-availability implementation for SAN connectivity only allows active paths to a LUN through a single filer; the cluster partner can only provide passive paths.
3. A filer can only be either the source of a synchronous SnapMirror relationship or the destination at one time in the current Data ONTAP version.

These limitations are discussed more fully below.

The operational issue surfaced when running significant I/O with SnapMirror in asynchronous mode. For example, when two hosts in each data center with two or more processor threads were running iometer or iorate to perform as many write I/Os as possible, replication either lagged so far behind that the data became significantly out of date on the target filer or replication aborted altogether, requiring a quiesce/break/resync command sequence to recover. This sequence discards changes made since the last time the source and target were in sync. When I/O is reduced to one host in each data center running a single I/O thread, replication lags an acceptable amount and does not abort. However, this amount of I/O is significantly less than what the other vendors could sustain. This issue was not seen with synchronous SnapMirror, although because of design limitation #3, only two test hosts were producing I/O for testing.

The filers were running version 7.2.2 of ONTAP to support the relatively new model X1124A-R6 FC-VI adapters (PCI Express, 4 Gbps capable). This is the first ONTAP version that supports these adapters. These new cards were not susceptible to the issue found in the older model X1024 FC-VI cards used in DCAP 2.0 testing. This issue caused link flaps unless I/O was throttled using a statement like this in the snapmirror.conf file:

fcip:async1 dcap-netapp-A1:async1 kbs=10000 * * * *

Note The "kbs=10000" limits the bandwidth utilization to 10,000 KB/sec. Unfortunately the issue with lagging or aborted replication is just as disruptive.

NetApp doesn't currently officially support asynchronous replication over FCIP (see http://now.netapp.com/NOW/knowledge/docs/switches/sm_fc_switch_support/index.shtml), DCAP 4.0 testing showed that it basically works, but only for relatively small traffic loads.

Here are the details of the design limitations that impact replication.

1. NetApp's IPFC and FC-VI-based implementation of SAN-based replication does not allow any benefit from write acceleration over either fiber channel or FCIP.

Neither FC nor FCIP write acceleration improves performance with NetApp SnapMirror replication. This is because of the FC-VI implementation, which doesn't use the fiber channel protocol (FCP) to send data. (FC-VI is a T11 standard for providing high-speed, low-latency interconnectivity for host clusters;

for more information see http://www.t11.org/.) This is true for both synchronous and asynchronous SnapMirror. DCAP testing showed that at least SnapMirror still works even though FC and FCIP write acceleration are enabled, although subject to the I/O amount constraints discussed earlier.

2. NetApp's high-availability implementation for SAN connectivity only allows active paths to a LUN through a single filer; the cluster partner can only provide passive paths.

The filers were configured in single system image mode, meaning the primary filer for the LUN presents an active path and the clustered partner filer presents a passive path to hosts. The NetApp Windows multipath driver (ONTAP DSM) and the Linux host attachment kits were used to make sure Windows 2003 and Linux hosts only used the active path unless it failed, in which case I/O would failover to the redundant, previously passive, path. Along with the Linux host attachment kit being installed, the following configuration was added to the /etc/multipath.conf file:

```
devices {
 device {
 vendor "NETAPP "
 product "LUN "
 path_grouping_policy group_by_prio
 getuid_callout "/sbin/scsi_id -g -u -s /block/%n"
 path_checker readsector0
 path_selector "round-robin 0"
 prio_callout "/opt/netapp/santools/mpath_prio_ontap /dev/%n"
 features "1 queue_if_no_path"
 failback immediate
```

3. A filer can only be either the source of a synchronous SnapMirror relationship or the destination at one time in the current Data ONTAP version.

Because of the limitation that a single filer can only be either the source or destination of a synchronous SnapMirror relationship, only two hosts (dcap-san-hst-01 and dcap-san-hst-04) had active synchronous SnapMirror-replicated LUNs.

Because each filer only has two FC-VI interfaces, one dedicated for FC and the other for FCIP, only one path in each direction is available for each type of replication. The FC paths used for synchronous replication had a theoretical bandwidth of 4 Gbps, and the FCIP paths used for asynchronous replication had a theoretical bandwidth of 311 Mbps (half an OC-12). For each I/O test, a Linux host and a Windows host were each performing 8 km writes in two processor threads as fast as possible in each data center to stress the replication topology.

The VSAN load balancing policy for port channels was SID/DID, and in-order delivery (IOD) was enabled for the NetApp replication VSANs.

Table A-4 summarizes the iometer and iorate results from representative base connectivity and synchronous and asynchronous replication tests.

Note These results are only valid to compare results with various MDS optimizations applied; they are not a reflection on the performance capabilities or quality of the storage arrays themselves.

Table A-4 Network Appliance FAS6070 Iometer/Iorate Results (average per device)

Traffic Type	Distance	I/O type	Host	I/O per sec	MB per sec
Synchronous replication without FC write acceleration	0 km	8 KB sequential writes	Linux	3043	23.8
			Windows	1253	9.8
	100 km		Linux	1815	14.2
			Windows	849	6.6
Synchronous replication with FC write acceleration	100 km		Linux	1832	14.3
			Windows	832	6.5
Asynchronous replication alone	0 km		Linux	2341	18.3
			Windows	2137	16.7
	100 km		Linux	2222	17.4
			Windows	2048	16.0
	5000 km		Linux	2066	16.1
			Windows	2140	16.7
Asynchronous replication with FCIP write acceleration	0 km		Linux	2220	17.3
			Windows	2054	16.1
	100 km		Linux	2107	16.5
			Windows	2132	16.7
	5000 km		Linux	2244	17.5
			Windows	2079	16.2
Asynchronous replication with FCIP compression	0 km		Linux	2155	16.8
			Windows	2164	16.9
	100 km		Linux	2152	16.8
			Windows	2177	17.0
	5000 km		Linux	2109	16.5
			Windows	2129	16.6
Asynchronous replication with FCIP encryption	0 km		Linux	2379	18.6
			Windows	1893	14.8
	100 km		Linux	2272	17.8
			Windows	2062	16.1
	5000 km		Linux	2215	17.3
			Windows	1974	15.4
Asynchronous replication with FCIP write acceleration, compression, and encryption	0 km		Linux	2149	16.8
			Windows	2214	17.3
	100 km		Linux	2365	18.5
			Windows	2178	17.0
	5000 km		Linux	2141	16.7
			Windows	2193	17.1

Table A-5 summarizes the Network Appliance FAS6070 software and firmware configuration information.

Table A-5 Network Appliance FAS6070 Software/Firmware Information

Software Component	Function	Location	Version
ONTAP	operating system	frame	7.2.2
MPIO (device-mapper-multipath)	multipath	Linux hosts	v0.4.5 (16/06, 2005)
ONTAP DSM	multipath	Windows hosts	3.0

Table A-6 summarizes the Network Appliance FAS6070 hardware configuration information.

Table A-6 Network Appliance FAS6070 Hardware Information

Hardware Component	Quantity	Comments
Frame	4	Serials 073252 and 073251 in DCA, 073250 and 073249 in DCB.
Memory	32768 MB	per frame
Disk	28 @ 133 GB	2 shelves @ 14 disks per frame
FC HBAs	2 @ 2 ports @ 4 Gbps	per frame, for base connectivity
FC-VI HBAs	1 @ 2 ports @ 2 Gbps	per frame, for replication; 4 Gbps capable but configured for 2 Gbps due to SSM module constraints

Hewlett Packard

This section has general and detailed information about the Hewlett Packard XP10000 frames used in the DCAP SAN topology. Following a brief general summary of results in Table A-7, Table A-8 has software and firmware information, Table A-9 has hardware information, and at the end is representative host device information showing redundant host paths and replicated devices.

General Summary

No issues were found in host connectivity or replication, but HP's implementation of Continuous Access XP Journal for asynchronous replication causes FCIP write acceleration not to improve performance. The reason is simply that Journal is based on reading the data rather than writing it (meaning the target frame controls the transfer by issuing reads to the source frame). Journal was tested in DCAP 4.0 instead of Continuous Access XP Asynchronous due to a recommendation from HP that Journal is more robust and typically is recommended to customers over Asynchronous (although some limited Asynchronous testing was done).

For load balancing and high availability, synchronous replication used two FC paths, each with a theoretical bandwidth of 4 Gbps, and asynchronous replication used four FCIP paths, each with a theoretical bandwidth of 311 Mbps (half an OC-12). For each I/O test, a Linux host and a Windows host were each performing 8 km writes in 2 separate processor threads as fast as possible in each data center to stress the replication topology.

The load balancing policy for port channels was SID/DID.

Table A-7 summarizes the iometer and iorate results from representative base connectivity and synchronous and asynchronous replication tests.

Note These results are only valid to compare results with various MDS optimizations applied; they are not a reflection on the performance capabilities or quality of the storage arrays themselves.

Table A-7 HP XP10000 Iometer/Iorate Results (average per device)

Traffic Type	Distance	I/O type	Host	I/O per sec	MB per sec
Synchronous replication without FC write acceleration	0 km	8 KB sequential writes	Linux	5788	45.2
			Windows	2924	23.0
	100 km		Linux	2254	17.6
			Windows	1131	8.8
Synchronous replication with FC write acceleration	100 km		Linux	2811	22.0
			Windows	1383	10.8
Asynchronous replication alone	0 km		Linux	3229	25.2
			Windows	2944	25.4
	100 km		Linux	3322	26.0
			Windows	2845	22.2
	5000 km		Linux	4088	31.9
			Windows	2037	15.9
Asynchronous replication with FCIP write acceleration	0 km		Linux	3258	25.4
			Windows	2962	23.1
	100 km		Linux	3514	27.5
			Windows	2928	22.9
	5000 km		Linux	4303	33.6
			Windows	1478	11.5
Asynchronous replication with FCIP compression	0 km		Linux	3358	26.3
			Windows	2967	23.2
	100 km		Linux	3496	27.4
			Windows	2802	21.9
	5000 km		Linux	4864	38.4
			Windows	1294	10.1
Asynchronous replication with FCIP encryption	0 km		Linux	3558	27.8
			Windows	2765	21.6
	100 km		Linux	3863	30.2
			Windows	2765	21.6
	5000 km		Linux	4268	33.4
			Windows	1833	14.4
Asynchronous replication with FCIP write acceleration, compression, and encryption	0 km		Linux	3452	27.0
			Windows	3045	23.7
	100 km		Linux	3509	27.5
			Windows	2992	23.4
	5000 km		Linux	4709	36.8
			Windows	1450	11.3

Table A-8 summarizes the HP XP10000 software and firmware configuration information.

Table A-8 HP XP10000 Software/Firmware Information

Software Component	Function	Location	Version
Microcode	operating system	frame	50-08-05
Command View XP AE Device Manager	configuration and monitoring, replication control	Windows host, frame service processor	5.1.0-00
service processor	operating system	frame	50-08-05/00
RMI Server	remote management	frame	04_08_00
MPIO (device-mapper-mulitpath)	multipath	Linux hosts	v0.4.5 (16/06, 2005)
HP MPIO DSM Manager	HP MPIO multipath management	Windows hosts	v2.00.00
HP MPIO Full Featured DSM for XP Disk Arrays	multipath	Windows hosts	v2.00.01

Table A-9 summarizes the HP XP10000 hardware configuration information.

Table A-9 HP XP10000 Hardware Information

Hardware Component	Quantity	Comments
Frame	2	Serials 82836, 82931
Cache	20 GB	per frame
Disk	60 @ 146 GB	per frame
Fiber host ports	8 ports @ 2 Gbps	per frame
Fiber replication ports	8 ports @ 2 Gbps	per frame

Quantum

This section has general and detailed information about the Quantum (formerly ADIC) Scalar i500 tape library and the Veritas NetBackup software used in the DCAP SAN topology. Following a brief general summary of results in Table A-10, Table A-11 has software and firmware information, Table A-12 has hardware information, and at the end is representative host device information showing host paths and devices.

General Summary

No issues were found in FCIP tape acceleration testing with the Quantum tape library. Results clearly showed that tape acceleration worked as expected at all distances for both reading and writing data. It also showed that hardware acceleration significantly improved throughput while software acceleration degraded throughput due to processing overhead.

A single path with a bandwidth of 155 Mbps (OC-3) was used. Two streams used this path, one for each data set.

The VSAN load balancing policy for port channels was OXID/SID/DID.

Table A-10 summarizes the throughput and time required to back up or restore the test data for the FCIP tape acceleration tests. At no time was tape drive compression used in testing, nor was the time taken to mount, unmount, or position tapes included in the timing and throughput calculations. The time column represents the elapsed time for the last stream to finish.

Note These results are only valid to compare results with various MDS optimizations applied; they are not a reflection on the performance capabilities or quality of the library or tape drives themselves.

Table A-10 Quantum Scalar i500 Backup/Restore Results

Traffic Type	Distance	I/O type	Host	Time (H:)MM:SS	MB per sec
Local Baseline	0 km	read	Linux	6:54	69.2
Remote Baseline	0 km	read	Linux	28:50	16.5
	100 km			29:51	15.8
	5000 km			5:33:45	3.0
FCIP tape acceleration enabled, no compression	0 km	read	Linux	28:04	17.1
	100 km			28:41	16.6
	5000 km			28:56	16.5
FCIP tape acceleration enabled, hardware compression	0 km	read	Linux	16:58	34.7
	100 km			16:49	34.7
	5000 km			18:12	30.3
FCIP tape acceleration enabled, software compression	0 km	read	Linux	1:13:04	6.5
	100 km			1:12:35	6.6
	5000 km			1:12:52	6.5
Local Baseline	0 km	write	Linux	5:48	88.3

Table A-10 Quantum Scalar i500 Backup/Restore Results

Traffic Type	Distance	I/O type	Host	Time (H:)MM:SS	MB per sec
Remote Baseline	0 km	write	Linux	28:23	16.9
	100 km			33:44	14.3
	5000 km			5:33:45	1.46
FCIP tape acceleration enabled, no compression	0 km	write	Linux	26:48	17.8
	100 km			29:11	16.3
	5000 km			27:55	17.0
FCIP tape acceleration enabled, hardware compression	0 km	write	Linux	12:19	38.2
	100 km			13:03	36.4
	5000 km			12:28	38.2
FCIP tape acceleration enabled, software compression	0 km	write	Linux	1:13:20	7.0
	100 km			1:13:33	6.5
	5000 km			1:13:08	6.5

Table A-11 summarizes the Quantum Scalar i500 software/firmware components, function, location, and version.

Table A-11 Quantum Scalar i500 Software/Firmware Information

Software Component	Function	Location	Version
firmware	control	library	320G.GS004
firmware	control	tape drive	64D0
sled boot version	control	tape drive	430A.GU001
sled drive version	control	tape drive	430A.GU001
Symantec NetBackup	media control, catalog	Linux backup hosts	6.0MP5

Table A-12 summarizes the Quantum Scalar i500 hardware components, quantity, and any comments.

Table A-12 Quantum Scalar i500 Hardware Information

Hardware Component	Quantity	Comments
Tape Library (30 slot)	1	Serial A0C0117003 in DCA.
Tape Drives	2	IBM Ultrium-TD3 LTO-3 serial numbers 1210212640 and 1210196088
Tape Media	15	HP C7973A LTO2 (800 GB compressed capacity, 400 GB uncompressed)

APPENDIX **B**

WAAS Implementation

Cisco Wide Area Application Services (WAAS) can be integrated anywhere in the network path. To achieve maximum benefits, however, optimum placement of the Wide Area Application Engines (WAE) devices between the origin server (source) and clients (destination) is essential.

Incorrect configuration and placement of the WAEs can lead not only to poorly performing applications, but, in some cases, network problems can potentially be caused by high CPU and network utilization on the WAEs and routers. For this phase of testing the Application Control Engine (ACE) was used for traffic interception, load-balancing, and redirection at the data center Aggregation Layer in DCa as well as WCCPv2 at the WAN edge in DCb.

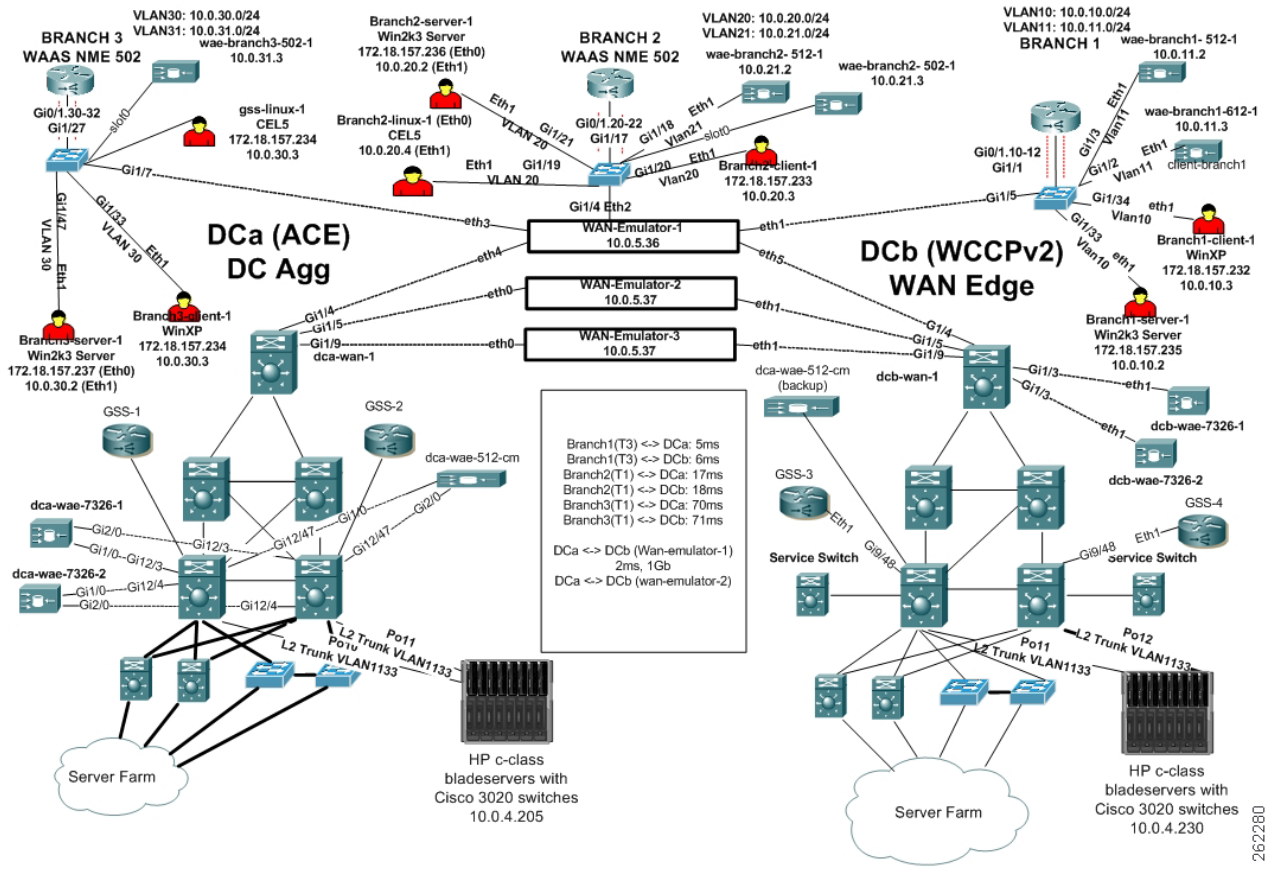
Design Components

For this Phase of DCAP testing a dual data center deployment with three remote branches was used. Both the functionality of the data center core and branch edge WAE devices and the redirection router/switches were tested. Ultimately the end user experience at each remote branch is the focus of this document. The key components of this WAAS design consist of the following:

- Cisco Catalyst 6500 with the ACE at the data center Aggregation Layer for WAAS packet interception and redirection
- Cisco Catalyst 6500 at the data center WAN edge, using WCCPv2 for WAAS packet interception and redirection
- Cisco WAE-7326 appliance(s) at the data center Aggregation Layer for Core WAAS services
- Cisco WAE-512 appliance(s) at the data center Aggregation Layer for Central Management
- Cisco 3845, 2821, or 2811 ISR at the branch/remote office for WCCPv2 packet interception
- Cisco WAE-612, WAE-512, and/or WAE-502 at the branch/remote office for WAAS termination

For this deployment, the Cisco Enterprise Solutions Engineering (ESE) design with the WAE's deployed at the data center Aggregation Layer with ACE redirection was used in DCa. The design guide for optimizing Oracle 11i E-Business Suite Traffic across the WAN with the ACE and WAAS solution was used as a basis for configuration in the data center with ACE. ESE design guides for WAAS at the WAN edge using WCCPv2 redirection were used to designate how the DCb configuration was done.

Figure B-1 WAAS ACE DCa and WCCPv2 DCb Topology



Data Center Core Details

The Cisco switch providing aggregation services for the data center was a Catalyst 6500 running Cisco Native IOS. A single Wide-Area Application Engine 512 (WAE-512) running Cisco (WAAS) was deployed in Central Manager mode and connected via Gigabit Ethernet copper to one of each data center's aggregation switches for management purposes and redundancy. For application acceleration purposes two WAE-7326s running Cisco WAAS was deployed in application-accelerator mode as a Core WAE and connected to each data center aggregation switch.

Remote Branch Details

The large, closely located (approximately 130km and 244km from DCa and DCb), branch office, appropriate for an office with up to 200 employees, connected to the WAN with a Cisco ISR 3845 running Cisco IOS was deployed with 5ms and 6ms of latency to DCa and DCb, respectively. The bandwidth of the large branch was restricted to that of a T3(45Mbps) connection. At this branch, a Cisco WAE-512 and a WAE-612 running Cisco WAAS were connected to the router and run in application-accelerator mode as Edge WAE.

The medium, distantly located (approximately 1134km and 1212km from DCa and DCb), branch office, appropriate for an office with up to 100 employees, connected to the WAN with a 2821 running Cisco IOS was deployed with 16ms and 17ms of latency to the data centers and the bandwidth was restricted to that of a T1(1.5Mbps). At this branch, a Cisco WAE-512 was connected via Gigabit Ethernet copper to the router and a NME-502 module was installed in an available slot within the router. Connectivity for the NME-502 came from the internal backplane connection to the router. Each WAE device was running in application-accelerator mode as an Edge device.

The small, distantly located (approximately 4559km and 4618km from DCa and DCb), branch office, appropriate for an office with up to 50 employees, connected to the WAN with a Cisco ISR 2811 running

Cisco IOS was deployed with 69ms and 70ms of latency and bandwidth restricted to that of a T1. A single WAE-502 was installed in the router and connected via the internal backplane connection.

Traffic Redirection Method

ACE

In data center A the ACE was used to redirect and load-balance traffic to the data center Core WAEs. WAE service devices deployed in the data center benefit from the availability and scalability services of the ACE platform. Each WAE was configured with a standby interface, connecting one of the physical interfaces to each aggregation switch. The ESE design guide for deploying Oracle 11i E-Business Suite with WAAS+ACE entitled “Optimizing Oracle E-Business Suite 11i across the WAN” was used and can be found at the following link:

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns483/c649/ccmigration_09186a0080857e32.pdf

The ACE was used to redirect and load-balance traffic to the data center Core WAEs. WAE service devices deployed in the data center benefit from the availability and scalability services of the ACE platform.

WCCPv2

In data center B the WAN edge router was configured to use WCCPv2 as the method for traffic interception and redirection. All incoming interfaces from the data center LAN were configured to redirect ingress traffic using WCCPv2 service group 62. The ingress WAN port was configured to redirect ingress traffic using WCCPv2 service group 61. The service group deployment scenario used is the most commonly deployed scenario.

Testing Concept

The WAAS+ACE testing focused on the basic network deployment, functionality and configuration for the WAE core devices and ACE module while using ACE redirection. For basic CIFS functionality from all three branches the WAFS Benchmark tool was used to verify that both optimization and file services were successful. In other scenarios a Shenick packet emulator was used to create truly stateful HTTP and FTP traffic. Performance improvements were evaluated during the testing, however, no stress testing was performed.

The WAAS+WCCPv2 testing focused on the basic network deployment, functionality and configuration for the 6500 router and WAE devices. Traffic generated by Ixia as well as the WAFS Benchmark tool to verify that both optimization and file services were successful.

APPENDIX **C**

Cisco GSS Implementation

The Global Site Selector (GSS) is a database of IP addresses, domain names, and resource records. The only sensitive data that resides on the GSS is the username and password used to access and make configuration changes. Content security is not a concern on the GSS. However, as an Administrator, one should be concerned with a few entities in regards to the DNS/security aspect. A DNS solution should not be vulnerable to a Distributed Denial of Service (DDoS) attack or allow unauthorized users access to the GSS.

Note The GSS does not run a distribution of BIND so it does not suffer from the many vulnerabilities that such open source code implementations are susceptible to.

A good understanding of the GSS's role in augmenting DNS and the requirements for accomplishing this task helps position the GSS properly in the network. Having an in-depth understanding of DNS and specifically how the GSS supports given sub-domains allows for effective positioning into existing network architectures.

The GSS may be deployed in a variety of locations in a customer's network to serve DNS requests, providing answers based on availability and preference. The GSS combines basic concepts of DNS with monitoring of answer status and providing users with IP addresses that are most appropriate for their content requests.

Design Components

A multi-data center deployment was designed and implemented for DCAP 4.0 testing and three remote branch locations were used for initiating client traffic. Each of the remote branches consisted of a client machine, local DNS name server and a branch edge WAE. The client machines were used to initiate various types of traffic destined for one of the data centers, the local name server was used to provide local name resolution for the client machines as well as provide NS Forwarding to all four GSS's, and the WAEs were used to provide an endpoint for application optimization.

The GSS devices use many TCP and UDP ports to communicate with each other and other devices on the networks. These ports must also be taken into consideration when determining where the GSS should be positioned in the network. The appropriate ports must be allowed thought firewalls/routers.

Two GSS's were installed at each data center. It is important to understand the ports/protocols used for the GSS's to communicate with each other.

Following are the ports/protocols are used to communicate with the GSS network:

GSSM-M to GSS

- TCP ports 2001 - 2009 using a secure session [RMI over SSL (bidirectional)]
- Used for updating configuration changes

GSS to GSSM-M

- TCP ports 3001 - 3009 configurable range of 30 - 4000 seconds

GSS to GSSM-M

- UDP:2000
- Auto detect that other GSS devices are online

All four GSS devices communicate with the CSM's in DCb and the ACEs in the DCa. This communication is necessary in order to understand the health and availability of the CSM and ACE devices in the global topology. This function is called a keepalive. The GSS supports the following keepalives;

- TCP—verify an open/close connection sequence - to terminate the sequence, the GSS can be configured to send a RST or a FIN. The TCP destination port is configurable.
- HTTP HEAD—verify the GSS is able to receive a "200 OK” response code from the target IP address. Host Tag, Destination Port, and Path are all configurable.
- ICMP—verify the GSS is able to receive ICMP replies from the target IP address.
- KAL-AP—Uses a UDP transport where the GSS integrates a CSS/CSM/ACE in order to obtain load information on a particular VIP/Vserver or specific rule. KAL-AP keepalive type can be configured for either KAL-AP by "TAG" or KAL-AP by VIP.

Implementation Details

Once you have configured your GSS devices to connect to the network and have created the logical resources (source address lists, domain lists, answers and answer groups, and DNS rules) required for global server load balancing, the steps necessary to integrate the GSS device into the network infrastructure and start delivery of user queries to the GSS can be undertaken. This starts with modifying the parent domain's DNS server to delegate parts of its name space to the GSS devices. In DCAP 4.0, this is performed from each branch name server. Each branch name server is configured to NS Forward DNS queries to each of the four GSS devices.

Delegation to GSS Devices

Modifying the DNS servers to accommodate your GSS devices involves the following steps:

1. Adding name server (NS) records to your DNS zone configuration file that delegates your domain
2. Adding "glue" address (A) records to the branch name name servers DNS zone configuration file maps the DNS name of each of your GSS devices to an IP address

The following sections will take a closer look into how the GSS devices were deployed, step by step, from an initial setup perspective.

GSSM-S, GSSM-M, and GSS

There are two GSS devices deployed at each data center. Each GSS is connected into each of the two Aggregation Layer switches at each data center. Of the two GSS devices at each data center, one is installed as a GSSM and the other is installed as a GSS. A total of four GSS devices are installed across both DCa and DCb.

Having a GSSM locally at each data center provides for redundancy in the event that one of the GSSMs goes offline at one of the data centers, the GSSM in the other data center will assume the role of Master GSSM. The promotion of a GSSM-S (standby GSSM) to a GSSM-M (master GSSM) is a manual process which involves accessing the GSSM via SSH or TELNET, logging in, and issuing the command, **gssm standby-to-primary**.

APPENDIX **D**

HP c-Class BladeSystem Implementation

For this phase of DCAP testing the HP c-Class BladeSystem was used in the data center topology to provide server platforms and network connectivity for devices running TIBCO RV, Exchange server 2003, and Oracle 11i E-Business Suite. This testing was complementary to the overall Oracle solution in that it focused primarily on LAN connectivity provided by the Cisco 3020 integrated switch blade. The purpose of this section is to outline the design goals and give an overview of the implementation of the HP BladeSystem as a whole.

Design Goals

- Provide a server platform for Oracle 11i E-Business Suite Application and Concurrent Management servers
- Consolidate and conserve server space, wiring, and power
- Provide a redundant connection to the data center LAN
- Provide easy deployment and simplified configuration infrastructure that promotes growth

The BladeSystem enclosures were racked in a Cisco 42U Rack and placed into each Data Center. In DCa Figure D-1, four Cisco 3020 ethernet switches and two Cisco 9124 Fiber Channel switches were installed in the enclosure. The integrated switches in the DCA enclosure were physically wired to each of the data center aggregation switches via Gigabit Ethernet copper to a Cisco WS-X6748-GE-TX linecard. For this phase of testing two of these switches were used to provide dual-homed connectivity to the data center LAN aggregation switches. NIC teaming/bonding was used on several of the application servers to provide redundancy at the server NIC layer.

In DCb Figure D-1 , four Ethernet Pass-thru and two Fiber Channel pass-thru modules were installed in the enclosure. The application servers in this enclosure were then physically connected into access layer.

Figure D-1 Data Center A (DCa) Topology

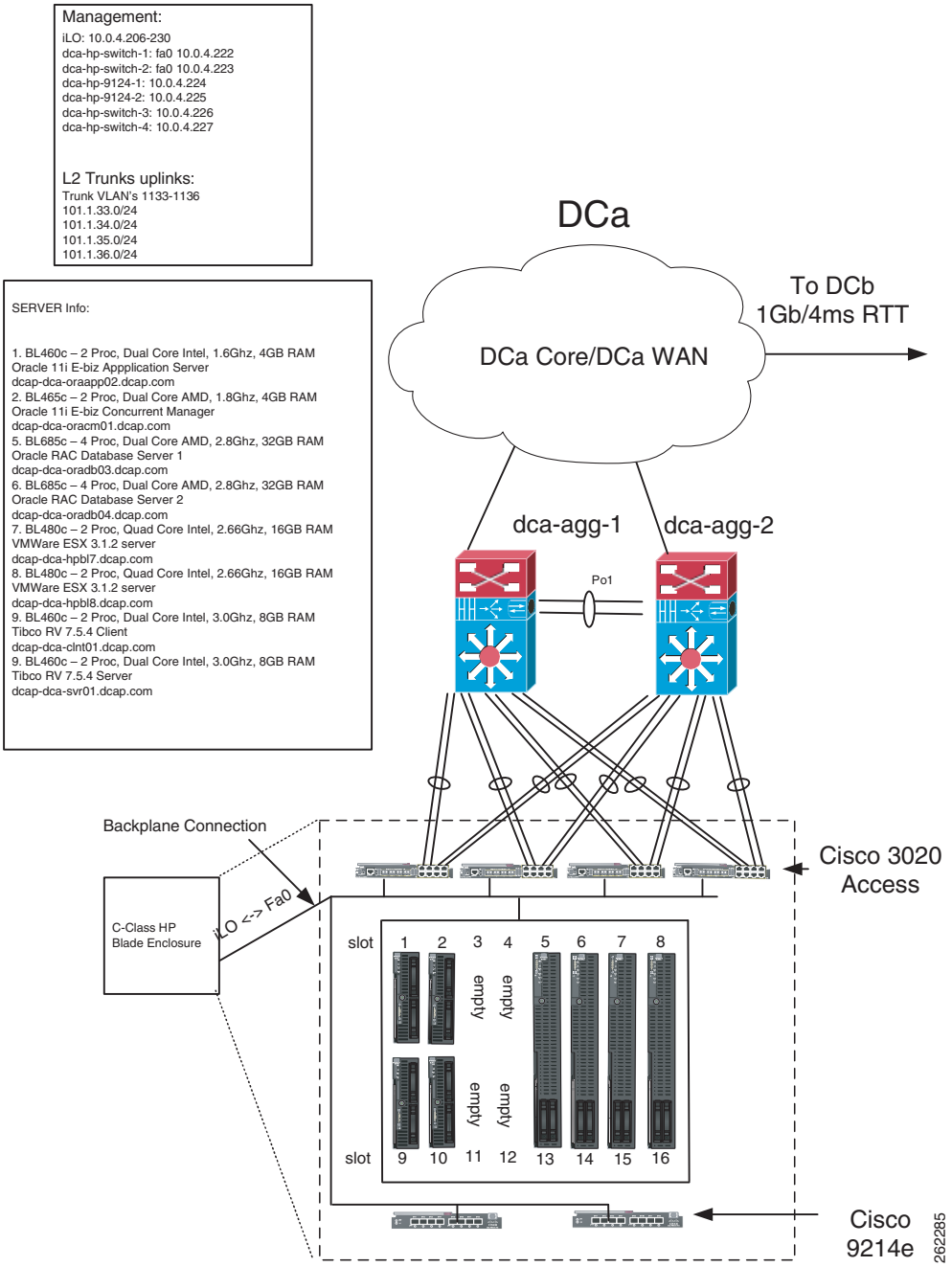
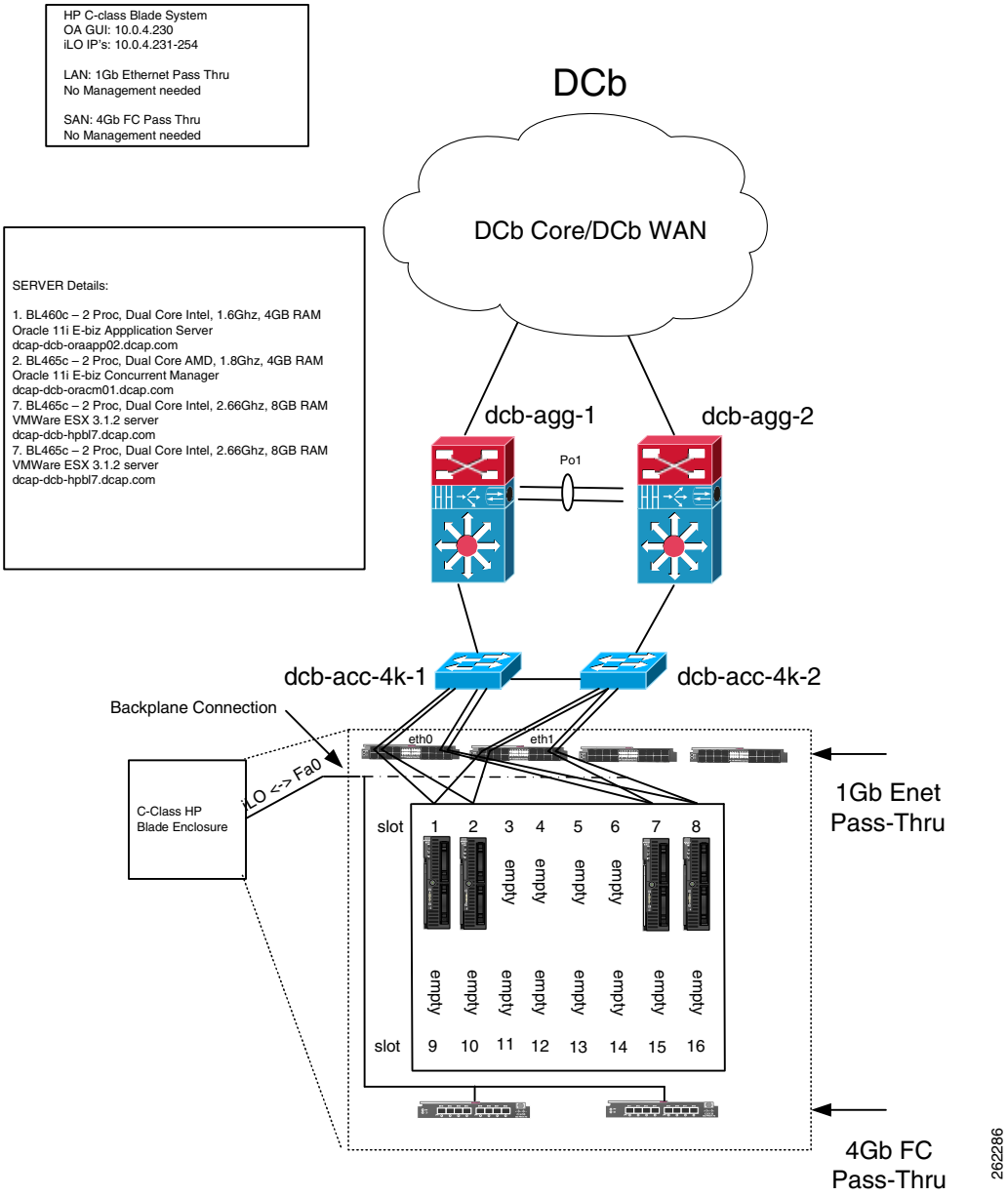


Figure D-2 Data Center B (DCb) Topology



Initial Configuration of HP Onboard Administrator

The Onboard Administrator (OA) for the c-class BladeSystem is used for managing the enclosure infrastructure. The first access available to the OA is through the enclosure's LCD HP Insight Display located on the front of the enclosure. This display is first used to configure the OA with an IP address, gateway, DNS, date and time, SNMP, and alert e-mail. The IP address and gateway are the only two necessary settings to configure to allow HTTPS browser access to the OA. Once a suitable IP address and Gateway are configured and the integrated Lights Out (iLO) port on the iLO blade on the back of the switch has been connected to the network, Internet Explorer 6.0 or higher can be used to access the web GUI.

Configuring Enclosure Bay IP Addressing

Devices can be accessed through the enclosure's iLO hardware interface which ties in to the HP OA. Each device can be administered an IP address which can be accessed by via the iLO port. Each iLO port can be accessed from an SSH terminal. To set this up you must first open up a browser to the HP OA GUI and navigate to the Enclosure Settings dropdown on the left sidebar, and finally to the link named Enclsure Bay IP Addressing (EBIPA). Clicking this link will bring up a form to provide static IP address assignment to the interconnect bays in the front and rear of the enclosure. DHCP is also an option but for this deployment static iLO IP addressing was optimal. Once an IP address is set for each system device, configuration and management of all devices can be done through the HP OA GUI.

Initial Configuration of Cisco 3020 Switch

There are two ways to access and configure the Cisco integrated switch, through the management console provided in the HP OA by the iLO, or through a physical management console port located on the Cisco switch. The first allows access to the device by browsing through the HP OA to the device located under the Interconnect Bays drop down menu on the left sidebar of the OA GUI. By selecting the device to configure the menu will further drop down providing accessibility to the software management console. Clicking the Management Console link will cause IE to open up a pop up window to the IP address of the device.

Once the device GUI has been opened the ability to configure the switch is available. Access to the traditional Cisco CLI via telnet is available through the GUI or through a terminal window from a host device a connected on a routable subnet. The second way to access the switch is to set up a console server and wire serial connections to the devices physical management console port located on the switch itself.

Accessing the devices through the HP OA GUI takes a few more steps than directly telnetting or consoling to the device, however, there are more features available through the GUI. For example, the ability to monitor bandwidth utilization, packet error and port utilization is available, along with, several ways to configure the switches that are simply unavailable via the CLI.

Configuring Cisco 3020 for Server to Network Connectivity

The Cisco 3020 should be configured to utilize a 2- or 4- port trunks to each of the Aggregation switches. In the DCAP setup, 2-port trunks were utilized and wired from the RJ-45 Gigabit Ethernet ports to a Cisco WS-X6748-GE-TX located in the data center aggregation layer switches. The trunking protocol used in this network was a basic PAgP, on-on, configuration. Once this trunk has been brought up between the Access and Aggregation layer the downlinks to the servers were configured to trunk the correct VLANs. Once configured as access switchport's, any server blade downlink port can connect to any other port on the switch including other servers within the enclosure (locally if they are in the same VLAN). Configuring all downlinks to use spanning-tree portfast is wise since there is no chance of creating a looped environment. When using multiple switches in the enclosure, multiple NIC's must be installed in the server in a 1:1 method. For example, if you have 4 switches then you will need 4 NIC's per server to connect to all 4 switches via the internal ports. Once you've set up the appropriate corresponding downlinks for switchport access and the uplink trunks to allow the necessary VLANs the servers are ready to be configured for in-band connectivity. At this point the servers must be assigned IP and gateway addresses appropriate for the VLAN that the server is to exist on.

Installing an Operating System on a Blade Server

As stated earlier, the iLO port address assigned by the EBIPA can be used to access any device within the chassis. Once a server has been physically installed in the enclosure and an IP address has been allocated for the iLO port, the OS can begin to be installed. In the OA browse to the Device Bay that the OS is to be loaded on and click the iLO link located under the correct device bay. Once the Device Bay screen loads, select the Web Administration link to access the iLO web user interface. For this phase of testing, two similar methods were used to load the servers with operating systems. Both required the mapping of a virtual drive from the local computer on which the OA was opened on. This mapping was done within the web administration page by clicking the Virtual Devices tab at the top and then clicking the Virtual Media tab on the left. A mapping to the local CDROM is available after opening the Virtual Media Applet and selecting the appropriate local drive. For Linux installations a bootable CD began the boot process which loaded the server to a point where a PXE boot could be used to continue the installation. The PXE was set up using the NIC1 MAC address as the hardware address to boot from. For Windows installations the media for the Windows OS was once again virtually mapped to the server from the local computer. The Windows installation was entirely loaded from the users local CDROM over the network.

Maintenance

Once the servers have been configured and network connectivity is established there is little maintenance necessary. Short of any hardware failure, the devices should stay in working order. In the case of a software failure or crash the devices are likely to come back up. If a circumstance arises and the server is unresponsive through the network the OA Integrated Remote Console can be used for direct access to the device even if it has failed to boot properly. In the case of a HP OA failure the iLO blade in the back of the enclosure can be reset by performing an OIR of the blade itself.

APPENDIX E

Oracle E-Business Configuration Details

The appendix has detailed Software and Hardware configuration information about the DCAP Oracle E-business suite Environment. Oracle Vision Demo environment is installed in multi-node mode using Oracle Installation tool “RAPID INSTALL”. 11.5.10.2 by default is installed with database verion 9iR2. The DB is upgraded to Oracle 10gR2 RAC and Oracle Applications are upgraded to latest Technology stack ATG.RUP4 patchset. Application tier and Database tier have been enabled with Autoconfig.

- Application URL: http://wwwin-oefin.gslb.dcap.com
- HTTP Port : 8000
- Forms Port : 9000

The following Oracle configuration details are available for Cisco DCAP 4.0.

Database and Application Configurations

The following Oracle software directory structure is used in Cisco DCAP 4.0:

```
Oracle Cluster Software :    /crs/oracle/product
Oracle RDBMS Software:  /oracle/product
Oracle ASM software:        /asm/oracle/product
```

Data Center A

The database OEFIN is configured in 2 node RAC cluster using 2 HP BL685’s . Oracle ASM is used for shared storage. Following sections details configuration files for ASM , Database,Listener and tnsnames.ora files.

Database and Application Tiers

The following Oracle database tier was was established for Cisco DCAP 4.0. Refer to Volume 25: Oracle E-Business Configurations for Database and Application tier configurations.

Hardware

- 2 HP BL685's with dual AMD opteron 2.8GHz CPU and 32GB of RAM in Dca

Hostnames

- dcap-dca-oradb03 ; VIP dcap-rac-node1.gslb.dcap.com
- dcap-dca-oradb04; VIP dcap-rac-node2.gslb.dcap.com
- 2 HP Proliant DL585 G2 with dual AMD opteron 2.8Ghz CPU and 32GB RAM in DCb

Software

- 10gR2 (10.2.0.3) RAC

Cluster

- Oracle CRS

OS

- Redhat Linux 4 update 4

Storage Vendors

- EMC, HP, and Netapp

CSM Configuration

The following CSM configuration is available for Cisco DCAP 4.0.

```
vserver WWWIN-OEFIN
  virtual 101.40.1.51 tcp 8000
  vlan 301
  serverfarm ORACLE-ALL
  advertise active
  sticky 30 group 34
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  domain dca-csm-1
  inservice
!
 vserver WWWIN-OEFIN-9K
  virtual 101.40.1.51 tcp 9000
  vlan 301
  serverfarm ORACLE-ALL
  advertise active
  sticky 30 group 2
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  domain dca-csm-1
  inservice
!
```

```
 vserver WWWIN-REDIRECT
  virtual 101.40.1.51 tcp www
  serverfarm 80-TO-8000
  persistent rebalance
  no inservice

probe ORACLE http
  credentials sysadmin  sysadmin
  header I_AM_CSM
  request method get url /oa_servlets/AppsLogin
  expect status 302
  interval 5
  failed 2
  port 8000

probe ORACLE-FORMS tcp
  interval 5
  retries 2
  port 9000


serverfarm ORACLE-ALL
  nat server
  nat client CLIENT_NAT
  real name DOT5
   inservice
  real name DOT16
   inservice
  probe ORACLE
  probe ORACLE-FORMS
```

GSS Configuration

The following GSS configuration is available for Cisco DCAP 4.0.

```
   Rule17:
      Name: wwwin-oefin
      Source Address List: Anywhere
      Domain List: wwwin-oefin.gslb.dcap.com
      Owner: System
      Status: Active
      Match DNS Query Type: A record
      Sticky Method: By Domain
      Sticky Inactivity Timeout: (global)
      Answer Group 1: 50-50-dca-and-dcb
      Balance Method 1: Round Robin
      Balance Clause Options 1: DNS TTL: 1; Return Record Count: 1; Sticky Enabl
e: Yes;
      Proximity Options 1: Proximity Enable: No
        RTT: (global)
        Zone: (global)
        Wait: (global)
      Answer Group 2:
      Balance Method 2:
      Balance Clause Options 2:
      Proximity Options 2:
      Answer Group 3:
      Balance Method 3:
      Balance Clause Options 3:
      Proximity Options 3:
Source Address Lists:
```

```
   List1:
      Name: Anywhere
      Owner: System
      Comments:
      Address Blocks: 0.0.0.0/0
   List2:
      Name: branch-1-src
      Owner: System
      Comments: DNS queries sourced from Branch 1
      Address Blocks: 10.0.10.0/24
   List3:
      Name: branch-2-src
      Owner: System
      Comments: DNS queries sourced from Branch 2
      Address Blocks: 10.0.20.0/24
   List4:
      Name: branch-3-src
      Owner: System
      Comments: DNS queries sourced from Branch 3
      Address Blocks: 10.0.30.0/24
   List5:
      Name: dca-campus-clients-src
      Owner: System
      Comments: DNS queries sourced from DCA
      Address Blocks: 101.1.34.0/24
   List6:
      Name: dcb-campus-clients-src
      Owner: System
      Comments: DNS queries sourced from DCB
      Address Blocks: 201.1.34.0/24
   List7:
      Name: list_1
      Owner: System
      Comments:
      Address Blocks: 1.1.1.1/32

   Answer11:
      IP Address: 101.1.33.12
      Name: SORRY-DCA
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:TCP to VIP Retries:1numProbs:1 KAL Destination Port: 80
        KAL Connection Termination Method: (global)
   Answer12:
      IP Address: 101.1.33.22
      Name: dca-mbox
      Status: Active
      Type: VIP
      Location:
      KeepAlives: Type:TCP to VIP Retries:(global)numProbs:(global) KAL Destinat
ion Port: 25
        KAL Connection Termination Method: Graceful
   Answer13:
      IP Address: 101.1.33.31
      Name: db-oefin-dca
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:TCP to VIP Retries:5numProbs:1 KAL Destination Port: 1531
        KAL Connection Termination Method: Graceful
   Answer14:
      IP Address: 101.1.33.32
      Name: db-crm-dca
```

```
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:ICMP to VIP Retries:(global)numProbs:(global)
   Answer15:
      IP Address: 101.1.33.34
      Name: wwwin-crm-dca
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:ICMP to VIP Retries:(global)numProbs:(global)
   Answer16:
      IP Address: 101.1.33.35
      Name: nas-oefin-dca
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:ICMP to VIP Retries:(global)numProbs:(global)   Type:TCP
to VIP Retries:(global)numProbs:(global) KAL Destination Port: 1531
        KAL Connection Termination Method: (global)
   Answer17:
      IP Address: 101.1.33.36
      Name: nas-crm-dca
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:ICMP to VIP Retries:(global)numProbs:(global)
   Answer18:
      IP Address: 101.1.33.50
      Name: wwwin-oefin-dca
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:HTTP HEAD to VIP Retries:(global)numProbs:(global) KAL Ho
st Tag:
        KAL Destination Port: 8000
        Path: (global)
        KAL Connection Termination Method: (global)
   Answer19:
      IP Address: 101.1.33.50
      Name: dcap-dca-oraapp01.dcap.com
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:HTTP HEAD to VIP Retries:(global)numProbs:(global) KAL Ho
st Tag:
        KAL Destination Port: 8000
        Path: (global)
        KAL Connection Termination Method: (global)
   Answer20:
      IP Address: 101.1.33.50
      Name: dcap-dca-oraapp02.dcap.com
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:HTTP HEAD to VIP Retries:(global)numProbs:(global) KAL Ho
st Tag:
        KAL Destination Port: 8000
        Path: (global)
        KAL Connection Termination Method: (global)
   Answer21:
      IP Address: 101.1.33.53
      Name: dcap-rac-node1
      Status: Active
```

```
      Type: VIP
      Location: dca
      KeepAlives: Type:TCP to VIP Retries:(global)numProbs:(global) KAL Destinat
ion Port: 1531
        KAL Connection Termination Method: (global)
   Answer22:
      IP Address: 101.1.33.54
      Name: dcap-rac-node2
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:TCP to VIP Retries:(global)numProbs:(global) KAL Destinat
ion Port: 1531
        KAL Connection Termination Method: (global)
   Answer23:
      IP Address: 101.1.33.59
      Name: dcap-mbox-2-DCA
      Status: Active
      Type: VIP
      Location: dca
      KeepAlives: Type:ICMP to VIP Retries:(global)numProbs:(global)
   Answer24:
      IP Address: 201.1.33.12
      Name: SORRY-DCB
      Status: Active
      Type: VIP
      Location: dcb
      KeepAlives: Type:TCP to VIP Retries:1numProbs:1 KAL Destination Port: 80
        KAL Connection Termination Method: (global)
   Answer25:
      IP Address: 201.1.33.22
      Name: dcb-mbox
      Status: Active
      Type: VIP
      Location:
      KeepAlives: Type:TCP to VIP Retries:(global)numProbs:(global) KAL Destination Port:
25
        KAL Connection Termination Method: Graceful
   Answer26:
      IP Address: 201.1.33.31
      Name: db-oefin-dcb
      Status: Active
      Type: VIP
      Location: dcb
      KeepAlives: Type:TCP to VIP Retries:5numProbs:1 KAL Destination Port: 1531
        KAL Connection Termination Method: Graceful
   Answer27:
      IP Address: 201.1.33.32
      Name: db-crm-dcb
      Status: Active
      Type: VIP
      Location: dcb
      KeepAlives: Type:ICMP to VIP Retries:(global)numProbs:(global)
   Answer28:
      IP Address: 201.1.33.34
      Name: wwwin-crm-dcb
      Status: Active
      Type: VIP
      Location: dcb
      KeepAlives: Type:ICMP to VIP Retries:(global)numProbs:(global)
   Answer29:
      IP Address: 201.1.33.35
      Name: nas-crm-dcb
      Status: Active
```

```
      Type: VIP
      Location: dcb
      KeepAlives: Type:TCP to VIP Retries:(global)numProbs:(global) KAL Destination Port:
8000
        KAL Connection Termination Method: (global)
   Answer30:
      IP Address: 201.1.33.35
      Name: nas-oefin-dcb
      Status: Active
      Type: VIP
      Location: dcb
      KeepAlives: Type:ICMP to VIP Retries:1numProbs:1   Type:TCP to VIP Retries
:1numProbs:1 KAL Destination Port: 1531
        KAL Connection Termination Method: (global)
   Answer31:
      IP Address: 201.1.33.53
      Name: dcap-rac-node1-dcb
      Status: Active
      Type: VIP
      Location: dcb
      KeepAlives: Type:TCP to VIP Retries:(global)numProbs:(global) KAL Destination Port:
1531
        KAL Connection Termination Method: (global)
   Answer32:
      IP Address: 201.1.33.53
      Name: dcap-rac-node2-dcb
      Status: Active
      Type: VIP
      Location: dcb
      KeepAlives: Type:TCP to VIP Retries:(global)numProbs:(global) KAL Destination Port:
1531
        KAL Connection Termination Method: (global)
   Answer33:
      IP Address: 201.1.33.59
      Name: dcap-mbox-2-DCB
      Status: Active
      Type: VIP
      Location: dcb
      KeepAlives: Type:ICMP to VIP Retries:(global)numProbs:(global)
   Answer34:
      IP Address: 201.40.30.51
      Name: wwwin-oefin-dcb
      Status: Active
      Type: VIP
      Location: dcb
      KeepAlives: Type:ICMP to VIP Retries:(global)numProbs:(global)
```

HP Load Runner Configurations

There were 5 business processes used for this testing cycle. Brief description of the steps involved for each of the business cases is outlined below.

- Business Test Case 1—CRM_Manage_Role, page E-8
- Business Test Case 2—iProcurement_Add_Delete_item, page E-8
- Business Test Case 3—Create_User, page E-8
- Business Test Case 4—Create_project_forms, page E-9
- Business Test Case 5—DCAP_Receivables, page E-9

Business Test Case 1—CRM_Manage_Role

Business test case 2 procedures are as follows:

Step 1 Go to homepage http://wwwin-oefin.gslb.dcap.com:8000/ and click "Apps Logon Link."

Step 2 Click on "ebusiness home page."

Step 3 Login using user id: sysadmin and password: sysadmin.

Step 4 Click: CRM HTML Administration.

Step 5 Click Setup : Home.

Step 6 Click Users.

Step 7 Click User Maintenance.

Step 8 Type lrt% and click go.

Step 9 Select User from the list for ex LoadTest1.

Step 10 Click on Roles.

Step 11 Click Update.

Step 12 Logout and Close all browsers.

Business Test Case 2—iProcurement_Add_Delete_item

Business test case 2 procedures are as follows:

Step 1 Go to homepage http://wwwin-oefin.gslb.dcap.com:8000/ and click "Apps Logon Link."

Step 2 Click on "ebusiness home page."

Step 3 Login using user id: sysadmin and password: sysadmin.

Step 4 Click: iProcurement.

Step 5 Click Categories.

Step 6 Click Ergonomic Supplies.

Step 7 Click Ankle Supports.

Step 8 Click Add to cart for Model 430.

Step 9 Click View Cart.

Step 10 Delete Item.

Step 11 Logout.

Business Test Case 3—Create_User

Business test case 3 procedures are as follows:

Step 1 Go to homepage : http://wwwin-oefin.gslb.dcap.com and click "Apps Logon Link."

Step 2 Click on "ebusiness home page."

Step 3 Login using User id: sysadmin and password sysadmin.

Step 4 Click: System Administrator.

Step 5 Click: Security : User : Define.

Step 6 Enter DCAPLR<sequence number generator> in the username field.

Step 7 Tab to password column and pass in "cisco" as password. password needs to be entered twice.

Step 8 Navigate to other tabs until you get to "Responsibility" column.

Step 9 Click List of values button and type in "System Administrator" and press "find" button. Select value.

Step 10 Click on "save" icon in the menu.

Step 11 Logout and close all browsers.

Business Test Case 4—Create_project_forms

Business test case 4 procedures are as follows:

Step 1 Go to homepage http://wwwin-oefin.gslb.dcap.com:8000/ and click “Apps Logon Link.”

Step 2 Click on “ebusiness home page.”

Step 3 Login using user name : sysadmin and password: sysadmin.

Step 4 Click project vision services -->project.

Step 5 Select project number: T, Cost Plus.

Step 6 Click find.

Step 7 Click copy to.

Step 8 Enter Project number: keep unique.

Step 9 Enter Name: ANY UNIQUE NAME.

Step 10 Select Project mgr: Marlin, Ms. Amy, Cust name: Hilman and Associates, project date: some future date (20-FEB-2007), end date: future date later then pro date (28-FEB-2007), Org: Vision Services R+D, PRODUCT: NON CLASSIFIED.

Step 11 Click OK: project gets created.

Step 12 Click open.

Step 13 Change status to approve.

Step 14 Logout and Close all browsers.

Business Test Case 5—DCAP_Receivables

Business test case 5 procedures are as follows:

Step 1 Go to homepage http://wwwin-oefin.gslb.dcap.com:8000/ and click “Apps Logon Link.”

Step 2 Click on “ebusiness home page.”

Step 3 Login using user id: sysadmin and password: sysadmin.

Step 4 Click: receivables vision operations.

Step 5 Click: transaction-->transactions.

Step 6 Select Source--?manual, class: invoice, ship to, sales person.

Step 7 Click on save – Message “transaction complete 1 record saved” is displayed at the bottom of the browser.

Step 8 Get invoice number which is dynamic.

Step 9 Enter line item.

Step 10 Choose item, UOM, Quantity, unit price and Taxcode.

Step 11 Click Save.

Step 12 Logout and Close all browsers.

Runtime settings

The following settings were used for Simultaneous, Concurrent users test and reliability test runs.

- Standard log (Send messages only when error occurs)
- Run vuser as a thread
- Step download time out: 300 seconds
- HTTP Request connect time out: 300 seconds
- HTTP Request receive time out: 300 seconds
- Browser emulation
  - Simulate new user for each iteration
  - Clear cache on each iteration
- Internet Protocol (Preferences)
  - DNS Caching – NO
  - Keep-alive http connections YES

Application NAS Details

The appl_top (/apps/oefin) volume shared by all 6 Application hosts was provided by a NetApp FAS6070 cluster in each data center. The hosts mounted the volume using NFSv2 over TCP.

Excerpt from /etc/fstab:

```
nas-oefin.gslb.dcap.com:/vol/dca_oraapp_oefin /apps/oefin nfs
nfsvers=2,tcp,rsize=32768,wsize=32768,hard,intr 0 0

[root@dcap-dca-oraapp01 ~]# /bin/mount | grep oefin
```

```
nas-oefin.gslb.dcap.com:/vol/dca_oraapp_oefin on /apps/oefin type nfs
(rw,nfsvers=2,tcp,rsize=32768,wsize=32768,hard,intr,addr=101.1.33.35)
```

The primary location was data center A and the failover location was data center B. NetApp synchronous SnapMirror replicated the data over IP. The WAN link used was accelerated by WAAS and WAFS.

Here are some details about the volume from the filer point of view:

```
dcap-netapp-A1> vol status dca_oraapp_oefin
         Volume State      Status            Options
dca_oraapp_oefin online     raid_dp, flex     nosnap=on, create_ucode=on,
                                              fs_size_fixed=on
                Containing aggregate: 'aggr1'

dcap-netapp-A1> snapmirror status dca_oraapp_oefin
Snapmirror is on.
Source                            Destination                    State         Lag
Status
dcap-netapp-A1:dca_oraapp_oefin   dcap-netapp-B1:dca_oraapp_oefin   Source       -
In-sync
```

Database Host Details

The data center A Oracle database servers are two HP BL685's with dual AMD opteron 2.8GHz CPU and32GB of RAM while the servers in data center B are two HP DL585's with dual AMD opteron 2.8GHz CPU and 32GB of RAM.

Each server boots from an internal RAID drive. Each server has 3 internal Broadcom GigE NICs, two for private network connectivity and one for test network connectivity. The two for private network are bonded to provide High Availability and used for cluster control traffic.

Servers dcap-dca-oradb03 and dcap-dca-oradb03 are in DCA and comprise the primary active/active Oracle database cluster. Servers dcap-dcb-oradb03 and dcap-dcb-oradb04 are in DCB and comprise the failover active/passive Oracle cluster.

All servers are running 64-bit RedHat Enterprise Linux 4 update 4, SMP kernel 2.6.9-42.EL. Servers in Dca are configured in Active/Active Real Application Cluster configuration using Oracle Clusterware while the servers in DCb are configured in Active/Passive cluster leveraging RedHat Cluster Suite.

Each database has 2 FC SAN ports provided by QLogic QLA2462 HBAs. The firmware version is 4.00.18 and the driver version is 8.01.04-d7.

The Oracle data was distributed over the 5 LUNs as follows:

Each cluster node had visibility to these files along with a 100 MB device for cluster quorum, control disks, and in some cases other disks that weren't used in testing.

EMC

```
DATA 1: /dev/mapper/SEMC_____SYMMETRIX______30032005B000p1 (67 GB) (shared, replicated to
DCB)

DATA 2: /dev/mapper/SEMC_____SYMMETRIX______300320033000p1 (67 GB) (shared, replicated to
DCB)

RECOVERY_AREA 1: /dev/mapper/SEMC_____SYMMETRIX______300320037000p1 (67 GB) (shared,
replicated to DCB)
```

```
RECOVERY_AREA 2: /dev/mapper/SEMC_____SYMMETRIX______30032007B000p1 (67 GB) (shared,
replicated to DCB)

REDO: /dev/mapper/SEMC_____SYMMETRIX______30032012F000p1 (1 GB) (shared, replicated to
DCB)
voting disk: /dev/mapper/360060e8014439400000143940000000e1 (10 GB, LUN 6) (shared)

/oracle: /dev/mapper/360060e8014439400000143940000000028p1 (100 MB, LUN 9)
OCR: /dev/mapper/360060e8014439400000143940000000dp1 (10 GB, LUN 5)(10 GB, LUN 5) (shared)

/oracle/admin: /dev/mapper/360060e8014439400000143940000000026p1 (20 GB, LUN 7)
```

HP

```
DATA 1: /dev/mapper/360060e8014439400000143940000000002p1 (71 GB, LUN 0) (shared, replicated
to DCB)

DATA 2: /dev/mapper/360060e8014439400000143940000000006p1 (71 GB, LUN 1) (shared, replicated
to DCB)

RECOVERY_AREA 1: /dev/mapper/360060e8014439400000143940000000a1 (71 GB, LUN 2) (shared,
replicated to DCB)

RECOVERY_AREA 2: /dev/mapper/360060e8014439400000143940000000020p1 (71 GB, LUN 3) (shared,
replicated to DCB)

REDO: /dev/mapper/360060e8014439400000143940000000c1 (10 GB, LUN 4) (shared, replicated to
DCB)

CRS and RDBMS code: /dev/mapper/360060e8014439400000143940000000049p1 and p2 (20 GB, LUN 8)

voting disk: /dev/mapper/360060e8014439400000143940000000e1 (10 GB, LUN 6) (shared)

/oracle: /dev/mapper/360060e8014439400000143940000000028p1 (100 MB, LUN 9)
```

Netapp

```
OCR: /dev/mapper/360a98000486e5365574a4744496b6656p1 (2 GB, LUN 5) (shared)

/oracle/admin: /dev/mapper/360a98000486e5365574a47444a306641p1 (22 GB, LUN 7)
/asm/oracle/product: /dev/mapper/360a98000486e5365574a47444a306641p2

DATA 1: /dev/mapper/360a98000486e5366535a4744494c6841p1 (75 GB, LUN 0) (shared, replicated
to DCB)

DATA 2: /dev/mapper/ 360a98000486e5366535a4744494d4d50p1 (75 GB, LUN 1) (shared,
replicated to DCB)

RECOVERY_AREA 1: /dev/mapper/360a98000486e5366535a4744494e6532p1 (75 GB, LUN 2) (shared,
replicated to DCB)

RECOVERY_AREA 2: /dev/mapper/360a98000486e5366535a4744494f496b1 (75 GB, LUN 3) (shared,
replicated to DCB)

REDO: /dev/mapper/360a98000486e5366535a474449516a68p1 (2 GB, LUN 4) (shared, replicated to
DCB)

CRS and RDBMS code: /dev/mapper/360a98000486e5365574a474449795764p1 and p2 (22 GB, LUN 8)
```

```
voting disk: /dev/mapper/360a98000486e5365574a4744496c5848p1 (2 GB, LUN 6) (shared)

/oracle: /dev/mapper/360a98000486e5365574a47444a357562p1 (130 MB, LUN 9)
```

Filesystems

Here's an excerpt from /etc/fstab that shows the storage for all three storage vendors (EMC,HP and Netapp) for one of the RAC nodes:

```
##EMC
#/dev/mapper/SEMC_____SYMMETRIX______30032011D000p1 /oracle ext3 _netdev 0 0
#/dev/mapper/SEMC_____SYMMETRIX______3003200ED000p1 /crs/oracle/product/ ext3 _netdev 0 0
#/dev/mapper/SEMC_____SYMMETRIX______3003200ED000p2 /oracle/product ext3 _netdev 0 0
#/dev/mapper/SEMC_____SYMMETRIX______3003200EF000p1 /oracle/admin ext3 _netdev 0 0
#/dev/mapper/SEMC_____SYMMETRIX______3003200EF000p2 /asm/oracle/product  ext3 _netdev 0 0

##HP
##/dev/mapper/360060e8014439400000143940000028p1 /oracle ext3 _netdev 0 0
##/dev/mapper/360060e8014439400000143940000049p1 /crs/oracle/product ext3 _netdev 0 0
##/dev/mapper/360060e8014439400000143940000049p2 /oracle/product ext3 _netdev 0 0
##/dev/mapper/360060e8014439400000143940000026p1 /oracle/admin ext3 _netdev 0 0
##/dev/mapper/360060e8014439400000143940000026p2 /asm/oracle/product  ext3 _netdev 0 0
##/dev/mapper/SEMC_____SYMMETRIX______300320073000p1 /oracle-emc/oradata/OEFIN/fs01 ext3
_netdev 0 0
##/dev/mapper/SEMC_____SYMMETRIX______300320077000p1 /oracle-emc/oradata/OEFIN/fs02 ext3
_netdev 0 0
##/dev/mapper/SEMC_____SYMMETRIX______30032007B000p1 /oracle-emc/archive/OEFIN/fs01 ext3
_netdev 0 0
##/dev/mapper/SEMC_____SYMMETRIX______3003200F8000p1 /oracle-emc/oradata/OEFIN/redo01 ext3
_netdev 0 0

##Netapp
/dev/mapper/360a98000486e5365574a47444a357562p1 /oracle ext3 _netdev 0 0
/dev/mapper/360a98000486e5365574a474449795764p1 /crs/oracle/product ext3 _netdev 0 0
/dev/mapper/360a98000486e5365574a474449795764p2 /oracle/product ext3 _netdev 0 0
/dev/mapper/360a98000486e5365574a47444a306641p1 /oracle/admin ext3 _netdev 0 0
```

Following information shows SAN Storage details for each of the vendor (EMC, HP and Netapp):

EMC

```
 [dcap-san-hst-06]# /usr/symcli/bin/symrdf -sid 320 -rdfg 17 -rdfa -nop -f
/devinfo/storage/emc/dcap-dca-oradb-rac_17_sync.rdf query

Symmetrix ID                       : 000190300320
Remote Symmetrix ID                : 000190300321
RDF (RA) Group Number              :  17 (10)

RDFA Session Number                : 16
RDFA Cycle Number                  : 0
RDFA Session Status                : Inactive
RDFA Minimum Cycle Time            : 00:00:30
RDFA Avg Cycle Time                : 00:00:00
Duration of Last cycle             : 00:00:00
RDFA Session Priority              : 33
Tracks not Committed to the R2 Side: 0
Time that R2 is behind R1          : 00:00:00
RDFA R1 Side Percent Cache In Use  :  0
```

```
RDFA R2 Side Percent Cache In Use  :  0
R1 Side DSE Used Tracks            : 0
R2 Side DSE Used Tracks            : 0
Transmit Idle Time                 : 00:00:00


       Source (R1) View                 Target (R2) View     MODES
-------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAC  STATE
-------------------------------- -- ------------------------ ----- ------------

N/A     0033 RW       0        0 RW 0033 WD       0        0 S...  Synchronized
N/A     0037 RW       0        0 RW 0037 WD       0        0 S...  Synchronized
N/A     005B RW       0        0 RW 005B WD       0        0 S...  Synchronized
N/A     012F RW       0        0 RW 012F WD       0        0 S...  Synchronized
N/A     007B RW       0        0 RW 007B WD       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

NETAPP

```
dcap-netapp-A1> vol status dca_rac_data
         Volume State      Status            Options
   dca_rac_data online     raid_dp, flex     nosnap=on, nosnapdir=on,
                                             create_ucode=on,
                                             fs_size_fixed=on
                Containing aggregate: 'aggr2'
dcap-netapp-A1> vol status dca_rac_redo
         Volume State      Status            Options
   dca_rac_redo online     raid_dp, flex     nosnap=on, nosnapdir=on,
                                             create_ucode=on,
                                             fs_size_fixed=on
                Containing aggregate: 'aggr1'
dcap-netapp-A1> vol status dca_rac_recovery
         Volume State      Status            Options
dca_rac_recovery online     raid_dp, flex     nosnap=on, nosnapdir=on,
                                             create_ucode=on,
                                             fs_size_fixed=on
                Containing aggregate: 'aggr0'

dcap-netapp-A1> lun show
        /vol/dca_rac_data/data1       75g (80530636800)   (r/w, online, mapped)
        /vol/dca_rac_data/data2       75g (80530636800)   (r/w, online, mapped)
        /vol/dca_rac_recovery/reco1     75g (80530636800)   (r/w, online, mapped)
        /vol/dca_rac_recovery/reco2     75g (80530636800)   (r/w, online, mapped)
        /vol/dca_rac_redo/redo1        2g (2147483648)    (r/w, online, mapped)

dcap-netapp-B1> snapmirror status
Snapmirror is on.
```

```
Source                          Destination                     State         Lag
Status
wan-emulator-1:dca_oraapp_oefin  dcap-netapp-B1:dca_oraapp_oefin  Snapmirrored   00:00:11
Idle
fc:dca_rac_data                 dcap-netapp-B1:dca_rac_data      Snapmirrored   -
In-sync
fc:dca_rac_recovery             dcap-netapp-B1:dca_rac_recovery  Snapmirrored   -
In-sync
fc:dca_rac_redo                 dcap-netapp-B1:dca_rac_redo      Snapmirrored   -
In-sync
```

HP

```
 [dcap-san-hst-10]# /usr/bin/pairdisplay -I0 -g dcap-dca-oradb-rac-s -fxce
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,   %,P-LDEV# M CTG JID AP
EM       E-Seq# E-LDEV#
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-0(L) (CL5-B , 0,  27)82836     2.P-VOL PAIR
NEVER ,  100     2 -   -   -  2  -            -       -
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-0(R) (CL5-B , 0,  25)82931     2.S-VOL PAIR
NEVER ,  100     2 -   -   -  -  -            -       -
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-1(L) (CL5-B , 0,  29)82836     6.P-VOL PAIR
NEVER ,  100     6 -   -   -  2  -            -       -
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-1(R) (CL5-B , 0,  27)82931     6.S-VOL PAIR
NEVER ,  100     6 -   -   -  -  -            -       -
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-2(L) (CL5-B , 0,  31)82836     a.P-VOL PAIR
NEVER ,  100     a -   -   -  2  -            -       -
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-2(R) (CL5-B , 0,  29)82931     a.S-VOL PAIR
NEVER ,  100     a -   -   -  -  -            -       -
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-3(L) (CL5-B , 0,  19)82836    20.P-VOL PAIR
NEVER ,  100    30 -   -   -  2  -            -       -
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-3(R) (CL5-B , 0,  37)82931    30.S-VOL PAIR
NEVER ,  100    20 -   -   -  -  -            -       -
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-4(L) (CL5-B , 0,  32)82836     c.P-VOL PAIR
NEVER ,  100    1a -   -   -  2  -            -       -
dcap-dca-oradb-rac-s    sync-dca-oradb-rac-4(R) (CL5-B , 0,  41)82931    1a.S-VOL PAIR
NEVER ,  100     c -   -   -  -  -            -       -
```

APPENDIX **F**

Exchange Configuration Details

This appendix has detailed hardware and configuration information about the DCAP Exchange environment.

Host Details

All four Exchange servers are HP Compaq DL380 G4s with 2 GB of RAM and 1 Intel Xeon 3.00 GHz CPU. Each server boots from an internal RAID drive.

Each primary server has 2 internal Broadcom GigE NICs in an active/passive NIC teaming configuration used for test network connectivity, and one additional NIC for private network connectivity. Each failover server has 2 internal Broadcom GigE NICs, one for private network connectivity and one for test network connectivity.

Servers dcap-xchng-a1 and dcap-xchng-a2 are in DCA and comprise the primary active/passive Exchange cluster. Servers dcap-xchng-b1 and dcap-xchng-b2 are in DCB and comprise the failover active/passive Exchange cluster.

All servers are running Microsoft Windows Server 2003 Enterprise Edition Service Pack 2.

All servers have Microsoft Exchange Server 2003 Enterprise Edition Service Pack 2.

Each has 2 FC SAN ports with either 2 QLogic QLE2460 or Emulex LP10000ExDC-E HBAs.

The break down is as follows:

- LP10000ExDC-E: dcap-xchng-a1, dcap-xchng-b1
  - Firmware: 1.91A5
  - Driver: 5-1.20A3
- LE2460: dcap-xchng-a2
  - Firmware: 4.00.23
  - Driver: 9.1.2.19 (w32)
- QLE2460: dcap-xchng-b2
  - Firmware: 4.00.17
  - Driver: 9.1.2.15 (w32)

Each server has 2 redundant FC paths to 3 data LUNs containing the Exchange database. There are 3 data LUNs per storage array (EMC DMX-3, HP XP10000, and NetApp FAS6070). The LUNs in DCA are synchronously replicated over a fiber channel link with 100 km of simulated distance to the LUNs in DCB.

The Exchange data was distributed over the 3 LUNs as follows:

- E: Exchange database files
- F: SMTP queue
- G: Exchange log files

Each cluster node had visibility to these files along with a 100 MB device for cluster quorum, control disks, and in some cases other disks that weren't used in testing.

The multipath software varied depending on the storage being tested. (Only one storage frame was visible from the host at any given time.) The break down is as follows:

- EMC: PowerPath 5.1.0 (build 51)
- NetApp: ONTAP DSM 3.0
- HP: HP MPIO Full Featured DSM for XP Disk Arrays v2.00.01 with HP MPIO DSM Manager v2.00.00

Windows Domain Controller Details

Each data center has a Domain Controller which doubles as a secondary DNS server and WINS server. The DC servers are running Windows Server 2003 Enterprise Edition SP1. The data center DC server receives zone file transfers from the master DNS server in the same data center only.

The domain is dcap.com (DCAP for pre-Windows 2000 hosts).

Each DC is configured as a Global Catalog Server.

DNS Details

Each data center has a master DNS Linux server configured to allow automatic DNS updates by Domain Controller servers and to disallow automatic updates fromWindows Exchange hosts. The reason for this is that otherwise when the Microsoft Exchange System Attendant resource for the Exchange Virtual Server is moved from one host to another, by default the new host updates DNS to reflect the new IP address. There is a setting to disable this in the network name resource for the Exchange Virtual Server, but as an added layer of security the DNS servers reject attempts from the Exchange hosts to update DNS.

The DNS servers are running RedHat Enterprise Linux version 4.4, kernel 2.6.9-42.7.ELsmp, 32-bit. Manual updates are made to both DNS servers at the same time. The reason for this is to facilitate data center failover (to keep from having to reconfigure the DNS server). The DNS servers are configured with a CNAME record for the dcap-mbox-1.dcap.com FQDN which refers the the canonical name dcap-mbox-1.gslb.dcap.com. This ensures GSS is authoritative for the canonical name so that site selection during disaster recovery failover and failback works properly. PTR records are also configured in the DNS servers to allow reverse lookup. Both the CNAME and PTR files are optional. Here are the CNAME and PTR records:

Zone file dcap.com

```
dcap-mbox-1          CNAME   dcap-mbox-1.gslb
```

Zone file 101.1.33.x

```
22              PTR     dcap-mbox-1.dcap.com.
```

Zone file 201.1.33.x

22 PTR dcap-mbox-1.dcap.com.

GSS Details

These are the configuration elements from the GSS that enabled GSS to provide site selection services for Exchange (extracted from the output of "show gslb-config"; for a complete listing, please see the GSS configuration appendix):

```
  domain-list dcap-mbox-2 owner System
        domain dcap-mbox-2.gslb.dcap.com

  domain-list mbox owner System
        domain dcap-mbox-1.gslb.dcap.com

  answer vip 101.1.33.22 name dca-mbox activate
        keepalive type tcp port 25 termination graceful ip-address 101.1.33.22


  answer vip 101.1.33.59 name dcap-mbox-2-DCA location dca activate
        keepalive type icmp ip-address 101.1.33.59


  answer vip 201.1.33.22 name dcb-mbox activate
        keepalive type tcp port 25 termination graceful ip-address 201.1.33.22

  answer vip 201.1.33.59 name dcap-mbox-2-DCB location dcb activate
        keepalive type icmp ip-address 201.1.33.59

  answer-group dcap-mbox owner System type vip
        answer-add 101.1.33.22 name dca-mbox weight 1 order 0 load-threshold 254 activate
        answer-add 201.1.33.22 name dcb-mbox weight 1 order 0 load-threshold 254 activate
  answer-group dcap-mbox-2-DCA owner System type vip
        answer-add 101.1.33.59 name dcap-mbox-2-DCA weight 1 order 0 load-threshold 254
activate
  answer-group dcap-mbox-2-DCB owner System type vip
        answer-add 201.1.33.59 name dcap-mbox-2-DCB weight 1 order 0 load-threshold 254
activate

  dns rule dcap-mbox-2 owner System source-address-list Anywhere domain-list dcap-mbox-2
query a

        clause 1 vip-group dcap-mbox-2-DCB method round-robin  ttl 5 count 1 sticky
disable
        clause 2 vip-group dcap-mbox-2-DCA method round-robin  ttl 5 count 1 sticky
disable


  dns rule mbox owner System source-address-list Anywhere domain-list mbox query a
        clause 1 vip-group dcap-mbox method round-robin  ttl 5 count 1 sticky disable
```

Each data center also has a Domain Controller which doubles as a secondary DNS server and WINS server. The DC servers are running Windows Server 2003 Enterprise Edition SP1. The data center DC server receives zone file transfers from the master DNS server in the same data center only.

Each of the three brances has a secondary DNS server. These servers are running Windows Server 2003 Enterprise Edition SP2. Each server is configured to receive zone file transfers from the master DNS servers in both data centers. The reason for this is to facilitate data center failover.

Storage Details

The following path and LUN details for dcap-xchng-a1 (other hosts are similar) are provided:

EMC

```
DISKPART> list disk

  Disk ###  Status      Size     Free     Dyn  Gpt
  --------  ----------  -------  -------  ---  ---
  Disk 0    Online        34 GB      0 B
  Disk 1    Online       119 MB      0 B      <<<< Q: clusters quorurm
  Disk 2    Online        67 GB      0 B      <<<< E: database
  Disk 3    Online        67 GB      0 B      <<<< F: SMTP queue
  Disk 4    Online        17 GB      0 B      <<<< G: logs

DISKPART> select disk 1

Disk 1 is now the selected disk.

DISKPART> detail disk

EMC SYMMETRIX  Multi-Path Disk Device
Disk ID: AC802D1C
Type   : FIBRE
Bus    : 0
Target : 0
LUN ID : 1

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 2     Q   Q            NTFS   Partition    119 MB  Healthy

DISKPART> select disk 2

Disk 2 is now the selected disk.

DISKPART> detail disk

EMC SYMMETRIX  Multi-Path Disk Device
Disk ID: AC802D3E
Type   : FIBRE
Bus    : 0
Target : 0
LUN ID : 2

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 3     E   xchng-emc-s  NTFS   Partition     67 GB  Healthy

DISKPART> select disk 3

Disk 3 is now the selected disk.

DISKPART> detail disk

EMC SYMMETRIX  Multi-Path Disk Device
Disk ID: AC802D3D
Type   : FIBRE
Bus    : 0
```

```
Target : 0
LUN ID : 3

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 4     F   xchng-emc-s  NTFS   Partition     67 GB  Healthy

DISKPART> select disk 4

Disk 4 is now the selected disk.

DISKPART> detail disk

EMC SYMMETRIX  Multi-Path Disk Device
Disk ID: AC802D13
Type   : FIBRE
Bus    : 0
Target : 0
LUN ID : 4

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 5     G   xchng-emc-s  NTFS   Partition     17 GB  Healthy


[DCAP-XCHNG-A1] C:\Program Files\EMC\PowerPath>powermt display dev=all
Pseudo name=harddisk1
Symmetrix ID=000190300320
Logical device ID=011B
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
---------------- Host ---------------   - Stor -   -- I/O Path -  -- Stats ---
### HW Path                I/O Paths    Interf.   Mode    State  Q-IOs Errors
==============================================================================
   2 port2\path0\tgt0\lun1     c2t0d1    FA  2cA   active  alive      0      0
   3 port3\path0\tgt0\lun1     c3t0d1    FA 15cA   active  alive      0      0

Pseudo name=harddisk2
Symmetrix ID=000190300320
Logical device ID=0063
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
---------------- Host ---------------   - Stor -   -- I/O Path -  -- Stats ---
### HW Path                I/O Paths    Interf.   Mode    State  Q-IOs Errors
==============================================================================
   2 port2\path0\tgt0\lun2     c2t0d2    FA  2cA   active  alive      0      0
   3 port3\path0\tgt0\lun2     c3t0d2    FA 15cA   active  alive      0      0

Pseudo name=harddisk3
Symmetrix ID=000190300320
Logical device ID=0067
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
---------------- Host ---------------   - Stor -   -- I/O Path -  -- Stats ---
### HW Path                I/O Paths    Interf.   Mode    State  Q-IOs Errors
==============================================================================
   2 port2\path0\tgt0\lun3     c2t0d3    FA  2cA   active  alive      0      0
   3 port3\path0\tgt0\lun3     c3t0d3    FA 15cA   active  alive      0      0

Pseudo name=harddisk4
Symmetrix ID=000190300320
Logical device ID=00F5
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
```

```
================================================================================
---------------- Host ----------------   - Stor -   -- I/O Path -  -- Stats ---
### HW Path                 I/O Paths    Interf.   Mode    State  Q-IOs Errors
================================================================================
   2 port2\path0\tgt0\lun4     c2t0d4    FA  2cA   active  alive      0      0
   3 port3\path0\tgt0\lun4     c3t0d4    FA 15cA   active  alive      0      0


[root@dcap-san-hst-06 ~]# /usr/symcli/bin/symrdf -sid 320 -rdfg 13 -nop -f /devi
nfo/storage/emc/dcap-dca-xchg_13_sync.rdf query

Symmetrix ID                       : 000190300320
Remote Symmetrix ID                : 000190300321
RDF (RA) Group Number              : 13 (0C)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDA   STATE
-------------------------------- -- ------------------------ ----- ------------

N/A     0063 RW       0        0 RW 0063 WD       0        0 S..   Synchronized
N/A     0067 RW       0        0 RW 0067 WD       0        0 S..   Synchronized
N/A     00F5 RW       0        0 RW 00F5 WD       0        0 S..   Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
```

NetApp

```
DISKPART> list disk

  Disk ###  Status      Size     Free     Dyn  Gpt
  --------  ----------  -------  -------  ---  ---
  Disk 0    Online        34 GB      0 B
  Disk 1    Online        40 GB      0 B
  Disk 2    Online        40 GB      0 B
  Disk 3    Online        40 GB      0 B
  Disk 4    Online       100 MB      0 B

DISKPART> select disk 1

Disk 1 is now the selected disk.

DISKPART> detail disk

NETAPP LUN  Multi-Path Disk Device
Disk ID: C8F444FA
Type   : FIBRE
Bus    : 0
```

```
Target : 0
LUN ID : 0

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 5     E   NETAPP SYNC  NTFS   Partition     40 GB  Healthy

DISKPART> select disk 2

Disk 2 is now the selected disk.

DISKPART> detail disk

NETAPP LUN  Multi-Path Disk Device
Disk ID: C8F44304
Type   : FIBRE
Bus    : 0
Target : 0
LUN ID : 1

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 3     F   NETAPP SYNC  NTFS   Partition     40 GB  Healthy

DISKPART> select disk 3

Disk 3 is now the selected disk.

DISKPART> detail disk

NETAPP LUN  Multi-Path Disk Device
Disk ID: C8F44307
Type   : FIBRE
Bus    : 0
Target : 0
LUN ID : 2

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 4     G   NETAPP SYNC  NTFS   Partition     40 GB  Healthy

DISKPART> select disk 4

Disk 4 is now the selected disk.

DISKPART> detail disk

NETAPP LUN  Multi-Path Disk Device
Disk ID: C8F44301
Type   : FIBRE
Bus    : 0
Target : 0
LUN ID : 3

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     U   Quorum       NTFS   Partition    100 MB  Healthy


[DCAP-XCHNG-A1] C:\Documents and Settings\plowden.DCAP>cd "c:\program files\netapp\mpio"

[DCAP-XCHNG-A1] C:\Program Files\NetApp\MPIO>dsmcli path list
Serial Number: HnSfSZBeW-I3
```

```
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x3000001
SCSI Address:
 Scsiport : 3
 HostPathId : 0
 Targetid : 0
 lun : 1
Path State:     ACTIVE

Dsm Id:         0x2000001
SCSI Address:
 Scsiport : 2
 HostPathId : 0
 Targetid : 0
 lun : 1
Path State:     PASSIVE

Serial Number: HnSfSZBeW942
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x3000000
SCSI Address:
 Scsiport : 3
 HostPathId : 0
 Targetid : 0
 lun : 0
Path State:     PASSIVE

Dsm Id:         0x2000000
SCSI Address:
 Scsiport : 2
 HostPathId : 0
 Targetid : 0
 lun : 0
Path State:     ACTIVE

Serial Number: HnSfSZBeWbhv
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x2000002
SCSI Address:
 Scsiport : 2
 HostPathId : 0
 Targetid : 0
 lun : 2
Path State:     ACTIVE

Dsm Id:         0x3000002
SCSI Address:
 Scsiport : 3
 HostPathId : 0
 Targetid : 0
 lun : 2
Path State:     PASSIVE

Serial Number: HnSfSZBeWdCr
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x3000003
```

```
SCSI Address:
 Scsiport : 3
 HostPathId : 0
 Targetid : 0
 lun : 3
Path State:     ACTIVE
Dsm Id:         0x2000003
SCSI Address:
 Scsiport : 2
 HostPathId : 0
 Targetid : 0
 lun : 3
Path State:     PASSIVE

dcap-netapp-A1> vol status EXCH_DB
         Volume State      Status            Options
        EXCH_DB online     raid_dp, flex     nosnap=on, create_ucode=on,
                                             fs_size_fixed=on
                Containing aggregate: 'aggr0'
dcap-netapp-A1> vol status EXCH_SMTP
         Volume State      Status            Options
      EXCH_SMTP online     raid_dp, flex     nosnap=on, create_ucode=on,
                                             fs_size_fixed=on
                Containing aggregate: 'aggr2'
dcap-netapp-A1> vol status EXCH_LOG
         Volume State      Status            Options
       EXCH_LOG online     raid_dp, flex     nosnap=on, create_ucode=on,
                                             fs_size_fixed=on
                Containing aggregate: 'aggr1'
dcap-netapp-A1> vol status EXCH_QUORUM
         Volume State      Status            Options
    EXCH_QUORUM online     raid_dp, flex     nosnap=on, create_ucode=on
                Containing aggregate: 'aggr2'


dcap-netapp-A1> snapmirror status EXCH_DB
Snapmirror is on.
dcap-netapp-A1:EXCH_DB      dcap-netapp-B1:EXCH_DB  Source         -          In-sync

dcap-netapp-A1> snapmirror status EXCH_SMTP
Snapmirror is on.
Source                        Destination               State          Lag        Status
dcap-netapp-A1:EXCH_SMTP      dcap-netapp-B1:EXCH_SMTP  Source         00:00:46   Idle


dcap-netapp-B1*> vol status EXCH_DB
         Volume State      Status            Options
        EXCH_DB online     raid_dp, flex     nosnap=on, snapmirrored=on,
                           snapmirrored      create_ucode=on,
                           read-only         fs_size_fixed=on,
                                             guarantee=volume(disabled)
                Containing aggregate: 'aggr0'
dcap-netapp-B1*> vol status EXCH_SMTP
         Volume State      Status            Options
      EXCH_SMTP online     raid_dp, flex     nosnap=on, snapmirrored=on,
                           snapmirrored      create_ucode=on,
                           read-only         fs_size_fixed=on,
                                             guarantee=volume(disabled)
                Containing aggregate: 'aggr2'
dcap-netapp-B1*> vol status EXCH_LOG
         Volume State      Status            Options
       EXCH_LOG online     raid_dp, flex     nosnap=on, snapmirrored=on,
                           snapmirrored      create_ucode=on,
                           read-only         fs_size_fixed=on,
```

```
                                          guarantee=volume(disabled)
                Containing aggregate: 'aggr1'
dcap-netapp-B1*> vol status EXCH_QUORUM
         Volume State      Status            Options
    EXCH_QUORUM online     raid_dp, flex     nosnap=on, create_ucode=on
                Containing aggregate: 'aggr2'


dcap-netapp-B1*> snapmirror status EXCH_DB
Snapmirror is on.
Source                Destination             State          Lag        Status
sonet:EXCH_DB         dcap-netapp-B1:EXCH_DB  Snapmirrored   -          In-sync
dcap-netapp-B1*> snapmirror status EXCH_SMTP
Snapmirror is on.
Source                Destination               State          Lag        Status
sonet:EXCH_SMTP       dcap-netapp-B1:EXCH_SMTP  Snapmirrored   -          In-sync
dcap-netapp-B1*> snapmirror status EXCH_LOG
Snapmirror is on.
Source                Destination              State          Lag        Status
sonet:EXCH_LOG        dcap-netapp-B1:EXCH_LOG  Snapmirrored   -          In-sync
```

HP

```
#   id        type,  bus,target,lun,vol_num,drive,vol_name
8, 67086708,  RAID,  0,  4,     0,  4,      C:
9,  3DDD63EF, FIBRE, 0,  1,     0,  5,      E:,   HP SYNC DB
10, 3DDD63F0, FIBRE, 0,  1,     1,  1,      F:,   HP SYNC SMTP
11, 3DDD63F2, FIBRE, 0,  1,     2,  2,      G:,   HP SYNC LOG
12, 3DDD63F4, FIBRE, 0,  1,     3
13, 3DDD63F6, FIBRE, 0,  1,     4
14, 3DDD63F8, FIBRE, 0,  1,     5,  3,      Q:,   Quorum


[DCAP-XCHNG-A1] C:\Program Files\Hewlett-Packard\HP MPIO DSM\XP DSM\x86>hpdsm devices

Device#  Device Name        Serial No.                      Active  Policy     Disk#
P.B.T.L
                                                                Paths
-----------------------------------------------------------------------------------------
-------------
 1       HP      OPEN-V      50 143940001                       2      NLB     Disk  9
2.0.1.0
 2       HP      OPEN-V      50 143940005                       2      NLB     Disk 10
2.0.1.1
 3       HP      OPEN-V      50 143940009                       2      NLB     Disk 11
2.0.1.2
 4       HP      OPEN-V-CM   50 143940069                       2      SQST    Disk 12
3.0.0.3
 5       HP      OPEN-V-CM   50 14394006A                       2      SQST    Disk 13
3.0.0.4
 6       HP      OPEN-V      50 1439400A5                       2      NLB     Disk 14
2.0.1.5

[DCAP-XCHNG-A1] C:\Program Files\Hewlett-Packard\HP MPIO DSM\XP DSM\x86>hpdsm paths
device=1
 Path#        Controller Port#           State        HBA Slot#   P.B.T.L

-------------------------------------------------------------------------
  1             5A                       Active           1       2.0.1.0 *
  2             6B                       Available        1       3.0.0.0
```

```
[DCAP-XCHNG-A1] C:\Program Files\Hewlett-Packard\HP MPIO DSM\XP DSM\x86>hpdsm paths
device=2

 Path#        Controller Port#           State        HBA Slot#   P.B.T.L
-------------------------------------------------------------------------
  1             5A                        Active           1       2.0.1.1 *
  2             6B                        Available        1       3.0.0.1

[DCAP-XCHNG-A1] C:\Program Files\Hewlett-Packard\HP MPIO DSM\XP DSM\x86>hpdsm paths
device=3

 Path#        Controller Port#           State        HBA Slot#   P.B.T.L
-------------------------------------------------------------------------
  1             5A                        Active           1       2.0.1.2 *
  2             6B                        Available        1       3.0.0.2


[root@dcap-san-hst-10 ~]# /usr/bin/pairdisplay -I0  -g dcap-dca-xchng-s -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-dca-xchng-s        sync-dca-xchng-e(L) (CL5-A , 0,   0)82836     1.P-VOL PAIR NEVER
,82931     1 -
dcap-dca-xchng-s        sync-dca-xchng-e(R) (CL5-A , 0,   0)82836     1.P-VOL PAIR NEVER
,82931     1 -
dcap-dca-xchng-s        sync-dca-xchng-f(L) (CL5-A , 0,   1)82836     5.P-VOL PAIR NEVER
,82931     5 -
dcap-dca-xchng-s        sync-dca-xchng-f(R) (CL5-A , 0,   1)82836     5.P-VOL PAIR NEVER
,82931     5 -
dcap-dca-xchng-s        sync-dca-xchng-g(L) (CL5-A , 0,   2)82836     9.P-VOL PAIR NEVER
,82931     9 -
dcap-dca-xchng-s        sync-dca-xchng-g(R) (CL5-A , 0,   2)82836     9.P-VOL PAIR NEVER
,82931     9 -
```

APPENDIX G

Disaster Recovery Configuration Details

This section provides detailed steps required to perform data center failover and failback.

Failover Procedure

The following procedure details steps required to perform data center failover.

Step 1 Start application transactions and verify environment.

1. Oracle: Begin sending Load Runner traffic and verify the application and network function properly.
2. Exchange: Verify Exchange application is running normally, then begin sending email from one Linux host in each DC and verify email reception for each branch server Outlook client (branch 1: testuser.00001, branch 2: testuser.00002, branch 3: testuser.00003).
3. Verify GSS.
4. Verify ACE.
5. Verify CSM.
6. Verify Load Runner.
7. Verify WAAS.
8. Verify NAS and SAN storage replication.

Step 2 Simulate a disaster situation in which all connectivity to DCa is terminated. Note the time.

Step 3 Fail over Oracle (database) and Exchange SAN storage.

EMC

On a solutions enabler host in DCb, do the following:

```
# /usr/symcli/bin/symrdf -sid 321 -rdfg 13 -nop -f
/devinfo/storage/emc/dcap-dca-xchg_13_sync.rdf failover

# /usr/symcli/bin/symrdf -sid 321 -rdfg 17 -nop -f
/devinfo/storage/emc/dcap-dca-oradb-rac_17_sync.rdf failover

# /usr/symcli/bin/symrdf -sid 321 -rdfg 13 -nop -f
/devinfo/storage/emc/dcap-dca-xchg_13_sync.rdf query
```

(check for a status of "Failed Over" or "Partitioned")

```
# /usr/symcli/bin/symrdf -sid 321 -rdfg 17 -rdfa -nop -f
/devinfo/storage/emc/dcap-dca-oradb-rac_17_sync.rdf query
```

(check for a status of "Failed Over" or "Partitioned")

Here are the configuration file contents:

```
/devinfo/storage/emc/dcap-dca-oradb-rac_17_sync.rdf:
 33 33
 37 37
 5b 5b
 12f 12f
 7b 7b

/devinfo/storage/emc/dcap-dca-xchg_13_sync.rdf
 63 63
 67 67
 f5 f5
```

HP

Using the HORCM CLI on the failover frame, run the following:

```
# /usr/bin/pairsplit -I0 -R -g dcap-dca-oradb-rac-s -vl
# /usr/bin/pairsplit -I0 -R -g dcap-dca-xchng-s -vl
# /usr/bin/pairdisplay -I0 -g dcap-dca-oradb-rac-s
```

(Check for status of "PSUE.")

```
# /usr/bin/pairdisplay -I0 -g dcap-dca-xchng-s -fxce
```

(Check for status of "PSUE.")

NetApp

On the failover filer, dcap-netapp-b1, do the following:

```
snapmirror quiesce dcap-netapp-B1:dca_rac_data
snapmirror break dcap-netapp-B1:dca_rac_data
snapmirror quiesce dcap-netapp-B1:dca_rac_recovery
snapmirror break dcap-netapp-B1:dca_rac_recovery
snapmirror quiesce dcap-netapp-B1:dca_rac_redo
snapmirror break dcap-netapp-B1:dca_rac_redo
snapmirror quiesce dcap-netapp-B1:EXCH_DB
snapmirror break dcap-netapp-B1:EXCH_DB
snapmirror quiesce dcap-netapp-B1:EXCH_SMTP
snapmirror break  dcap-netapp-B1:EXCH_SMTP
snapmirror quiesce dcap-netapp-B1:EXCH_LOG
snapmirror break dcap-netapp-B1:EXCH_LOG
```

Step 4 Fail over Oracle (application) NAS storage.

On filer dcap-netapp-b1, do "snapmirror quiesce dca_oraapp_oefin" then "snapmirror break dca_oraapp_oefin".

Step 5 Bring up Exchange database on the failover cluster and verify all branch clients can receive email.

Note the time when the first branch client can receive email (this is the Exchange Recovery Time Objective or RTO). Also note the time when all clients can receive email. Also verify how much data (email) if any, was lost (this is the Exchange Recovery Point Objective or RPO). Should be no data loss.

1. On a domain controller in the failover data center, reset the Exchange Virtual Server domain account.
2. On the primary fail over cluster node, create as needed and online the Exchange service group in Cluster Administrator; be sure to enable Kerberos and check the "ensure DNS changes succeed box" in the network name resource before onlining.
3. Still on the primary fail over cluster node, create a Microsoft Exchange System Attendant resource as needed which depends on the network name and the disk resources. Then again online the Exchange service group in Cluster Administrator.
4. On the primary fail over cluster node, using Exchange System Manager, mount both the mailbox and public folders stores as needed, and ensure the HTTP and SMTP protocols are using the DCb address (Properties). If you have to change the IP, offline and then online the resource in Cluster Administrator.
5. Verify DNS points to the correct address for the Exchange Virtual Server on the master DNS server in the failover data center. If not, fix it.
6. Verify DNS points to the correct address on all three branch DNS secondary servers.
7. Ensure the branch Outlook users begin to receive email again. Check for email until the queues are clear on both sending hosts. Flush the email queues on the hosts sending the emails as needed ("sendmail -q" to flush, "mailq" to check).
8. Verify email reception.

Step 6 Bring up Oracle database and listener on the failover cluster.

Make sure all required file systems are mounted, then issue the following commands to start the database and the listener.

1. Set the environment for the database.
2. sqlplus / **as sysdba**; startup
3. **./addlnctl.sh** start

Step 7 Verify GSS is giving out correct addresses for Oracle database, NAS, and clients. Then remount NAS storage on Oracle application nodes (may require reboot).

The command to run on the web application hosts is **mount /apps/oefin**.

Step 8 Bring up Oracle on the failover nodes, verify CSM, and verify GSS is directing all clients to DCb.

Note the time when the first branch client can access Oracle (this is the Oracle Recovery Time Objective or RTO). Also note the time when all clients can access Oracle. Also verify how much data, if any, was lost (this is the Oracle Recovery Point Objective or RPO). Should be no data loss.

1. Make sure NAS shared filesystem is mounted on all the Application hosts. Issue the command **df -k** to validate the filesystem availability.
2. Start the Application services from all the application hosts. Login to Application host and go to dir /apps/oefin/common_top/admin/scripts/.
3. Run the command **./adstrtal.sh apps/apps** .
4. Verify the log file created from the step to ensure all the required services are started successfully.
5. Login to the Application User screen and query for the last user created. This should show the user created prior to losing DC connectivity.

6. Submit the Application traffic through LR controller to ensure Transactions are submitted successfully in the failed over Datacenter.

Step 9 Determine the latest RTO and RPO of all applications.

This is the data center failover RTO and RPO.

Failback Procedure

The following procedure details steps required to perform data center failback.

Note This procedure assumes failover has already been done.

Step 1 Start application transactions and verify environment.

1. Oracle: Begin sending Load Runner traffic and verify the application and network is functioning properly.
2. Exchange: Verify Exchange application is running normally, then begin sending email from one Linux host in each DC and verify email reception for each branch server Outlook client (branch 1: testuser.00001, branch 2: testuser.00002, branch 3: testuser.00003).
3. Verify GSS.
4. Verify CSM.
5. Verify ACE.
6. Verify Load Runner.
7. Verify WAAS.
8. Verify NAS and SAN storage replication.

Step 2 Ensure the primary data center storage array is in the proper state, then restore SAN extension connectivity to the primary data center.

For HP storage, begin the resync of changes to DCa:

DCa host:

```
[dcap-san-hst-10]# /usr/bin/pairsplit -I0 -S -g dcap-dca-oradb-rac-s
[dcap-san-hst-10]# /usr/bin/pairsplit -I0 -S -g dcap-dca-xchng-s
```

DCb host:

```
[dcap-san-hst-12]# /usr/bin/paircreate -I0 -g dcap-dca-oradb-rac-s -f never -vl
[dcap-san-hst-12]# /usr/bin/paircreate -I0 -g dcap-dca-xchng-s -f never -vl
```

Step 3 Ensure the primary datacenter applications, including the ACE VIP for Oracle, are offline, then restore WAN connectivity to DCa.

Step 4 After the failback outage window begins, take all applications offline in the failover data center. Note the time.

For Oracle, shutdown the Application services and then shutdown the Listener and Database. Be sure to unmount NAS and SAN storage on DCb hosts.

For Exchange, on the primary node on the failover cluster, dismount the stores in Exchange System Manager, offline the Exchange service group in Cluster Administrator, and optionally delete the Microsoft Exchange System Attendant resource (do *not* remove the Exchange Virtual Server).

Step 5 After all applications are offline in the failover data center, fail back Oracle (database) and Exchange SAN storage.

EMC

On a solutions enabler host in DCA, do the following:

```
# /usr/symcli/bin/symrdf -sid 321 -rdfg 13 -force -nop -f
/devinfo/storage/emc/dcap-dca-xchg_13_sync.rdf failback
# /usr/symcli/bin/symrdf -sid 321 -rdfg 17 -force -nop -f
/devinfo/storage/emc/dcap-dca-oradb-rac_17_sync.rdf failback
/usr/symcli/bin/symrdf -sid 321 -rdfg 13 -rdfa -nop -f
# /devinfo/storage/emc/dcap-dca-xchg_13_sync.rdf query
```

(Look for status "Synchronized").

```
# /usr/symcli/bin/symrdf -sid 321 -rdfg 17 -rdfa -nop -f
/devinfo/storage/emc/dcap-dca-oradb-rac_17_sync.rdf query
```

(Look for status "Synchronized").

This method allows only changes made in DCb to be copied back to DCa.

Here are the configuration file contents:

```
/devinfo/storage/emc/dcap-dca-oradb-rac_17_sync.rdf:
  33 33
  37 37
  5b 5b
  12f 12f
  7b 7b

 /devinfo/storage/emc/dcap-dca-xchg_13_sync.rdf
  63 63
  67 67
  f5 f5
```

HP

Using the HORCM CLI on the failover frame, issue the following commands:

On DCb Host:

```
# /usr/bin/pairsplit -I0 -rw -g dcap-dca-oradb-rac-s
# /usr/bin/pairsplit -I0 -rw -g dcap-dca-xchng-s
# /usr/bin/pairdisplay -I0 -g dcap-dca-oradb-rac-s -fxce
```

(Check for status of "PSUS".)

```
# /usr/bin/pairdisplay -I0 -g dcap-dca-xchng-s -fxce
```

(Check for status of "PSUS".)

NetApp

On filer dcap-netapp-a1 (original source filer), do:

```
  snapmirror resync -f -S dcap-netapp-b1-e0c:dca_rac_data -w dcap-netapp-A1:dca_rac_data
  snapmirror break dcap-netapp-A1:dca_rac_data
  snapmirror resync -f -S dcap-netapp-b1-e0c:dca_rac_recovery -w
dcap-netapp-A1:dca_rac_recovery
  snapmirror break dcap-netapp-A1:dca_rac_recovery
  snapmirror resync -f -S dcap-netapp-b1-e0c:dca_rac_redo -w dcap-netapp-A1:dca_rac_redo
  snapmirror break dcap-netapp-A1:dca_rac_redo
  snapmirror resync -f -S dcap-netapp-b1-e0c:EXCH_DB -w dcap-netapp-A1:EXCH_DB
  snapmirror break dcap-netapp-A1:EXCH_DB
```

```
snapmirror resync -f -S dcap-netapp-b1-e0c:EXCH_SMTP -w dcap-netapp-A1:EXCH_SMTP
snapmirror break dcap-netapp-A1:EXCH_SMTP
snapmirror resync -f -S dcap-netapp-b1-e0c:EXCH_LOG -w dcap-netapp-A1:EXCH_LOG
snapmirror break dcap-netapp-A1:EXCH_LOG
```

This method allows only changes made in DCb to be copied back to DCa.

Step 6 Fail back Oracle (application) NAS storage.

On filer dcap-netapp-a1 (original source filer), do **snapmirror resync -f -S dcap-netapp-b1-e0c:dca_oraapp_oefin -w dcap-netapp-A1:dca_oraapp_oefin**. Then do **snapmirror break dcap-netapp-A1:dca_oraapp_oefin**. This method allows only changes made in DCb to be copied back to DCa.

Step 7 Bring up Exchange database on the primary cluster and verify all branch clients can receive email.

Note the time when the first branch client can receive email (this is the Exchange Recovery Time Objective or RTO). Also note the time when all clients can receive email. Also verify how much data (email) if any, was lost (this is the Exchange Recovery Point Objective or RPO). Should be no data loss.

After the outage window starts, bring up Exchange on the primary cluster:

1. On a domain controller, reset the domain account for the Exchange Virtual Server.
2. On the primary cluster node, create as needed and online the Exchange service group in Cluster Administrator; be sure to enable Kerberos and check the "ensure DNS changes succeed box" in the network name resource before onlining.
3. As needed, on the primary cluster node, create a Microsoft Exchange System Attendant resource. This resource should depend on the network name resource and the disk resources. Then, using Cluster Administrator, online the Exchange service group in the primary Exchange cluster. Finally, using Exchange System Manager, mount both the mailbox and public folders stores as needed, and verify the HTTP and SMTP protocols are using the proper address.
4. Verify DNS points to the correct address on all three branch DNS secondary servers.
5. Ensure the branch Outlook users begin to receive email again. Check for email until the queues are clear on both sending hosts.
6. Verify no email is lost.

Step 8 Bring up Oracle database and listener on the primary cluster.

Make sure all required file systems are mounted, then issue the following commands to start the database and the listener.

1. Set the environment for the database.
2. sqlplus / **as sysdba**; startup
3. **./addlnctl.sh** start

Step 9 Verify GSS is giving out correct addresses for Oracle database, NAS, and clients. Then remount NAS storage on Oracle application nodes (may require reboot).

The command to run on the web application hosts is **mount /apps/oefin**.

Step 10 Bring up Oracle application on the all Application nodes in both data centers, verify CSM, and verify GSS is loadbalancing clients to both DCa and DCb.

Note the time when the first branch client can access Oracle (this is the Oracle Recovery Time Objective or RTO). Also note the time when all clients can access Oracle. Also verify how much data, if any, was lost (this is the Oracle Recovery Point Objective or RPO). Should be no data loss.

1. Make sure NAS shared filesystem is mounted on all the Application hosts. Issue the command df -k to validate the filesystem availability.

2. Start the Application services from all the application hosts. Login to Application host and go to dir /apps/oefin/common_top/admin/scripts/.
3. Run the command ./adstrtal.sh apps/apps.
4. Verify the log file created from the step to ensure all the required services are started successfully.
5. Login to the Application User screen and query for the last user created. This should show the user created prior DC failback.
6. Submit the Application traffic through LR controller to ensure Transactions are submitted successfully and are loadbalanced across both DCa & DCb.

Step 11 Reinstate DCa to DCb replication for both SAN and NAS storage.

Reinstate DCa to DCb NAS replication by doing the following on filer dcap-netapp-b1 (can be done anytime): **snapmirror resync -f -w dcap-netapp-B1:dca_oraapp_oefin** and **snapmirror release dca_oraapp_oefin dcap-netapp-A1:dca_oraapp_oefin**.

SAN Storage

EMC

No additional work is needed.

HP

Using HORCM CLI on the DCb host, issue the following:

```
[dcap-san-hst-12]# /usr/bin/pairresync -I0 -swapp -g dcap-dca-xchng-s
[dcap-san-hst-12]# /usr/bin/pairresync -I0 -swapp -g dcap-dca-oradb-rac-s
[dcap-san-hst-12]# /usr/bin/pairdisplay -I0 -g dcap-dca-xchng-s –fxce
```

(Look for a status of "PAIR" or "COPY" and eventually just for "PAIR.")

```
[dcap-san-hst-12]# /usr/bin/pairdisplay -I0 -g dcap-dca-oradb-rac-s -fxce
```

(Look for a status of "PAIR" or "COPY" and eventually just for "PAIR.")

NetApp

Do the following on filer dcap-netapp-b1:

```
snapmirror resync -f -w dcap-netapp-B1:EXCH_DB
snapmirror release EXCH_DB dcap-netapp-A1:EXCH_DB
snapmirror resync -f -w dcap-netapp-B1:EXCH_SMTP
snapmirror release EXCH_SMTP dcap-netapp-A1:EXCH_SMTP
snapmirror resync -f -w dcap-netapp-B1:EXCH_LOG
snapmirror release EXCH_LOG dcap-netapp-A1:EXCH_LOG
snapmirror resync -f -w dcap-netapp-B1:dca_rac_data
snapmirror release dca_rac_data dcap-netapp-A1:dca_rac_data
snapmirror resync -f -w dcap-netapp-B1:dca_rac_recovery
snapmirror release dca_rac_recovery dcap-netapp-A1:dca_rac_recovery
snapmirror resync -f -w dcap-netapp-B1:dca_rac_redo
snapmirror release dca_rac_redo dcap-netapp-A1:dca_rac_redo
```

Step 12 Determine the latest RTO and RPO of all applications.

This is the data center failover RTO and RPO.

APPENDIX **H**

The Voodoo Solution

The DCAP test topology is highly scaled, in terms of the number of servers that are present across all serverfarms. In DCa, the CSM is configured with 2000+ real servers across 30+ serverfarms. Obviously, it is not realistic or cost-effective to deploy 2000 physical servers in a test environment like DCAP.

Emulating 2000 Servers in DCAP

One of the goals of the DCAP project is to stress the Catalyst 6500 and Catalyst 4900 products in their roles as access switches by fully populating one of each chassis with Fast Ethernet-connected servers. For a modular chassis such as the Catalyst 6509, this meant 6 48-port linecards, or 288 connected servers. This does not scale, either.

What is Voodoo?

The solution in DCAP testing used ten physical servers to emulate the 2000 that were configured in the LAN topology of DCa. These servers, combined with a Catalyst 6513 chassis that is separate from the DCa LAN infrastructure, provided the magic, or “Voodoo” that made this solution happen.

Why the Need for Voodoo?

So the problem was how to scale the number of real servers that the load-balancer could send traffic to as well as scale the functional connectivity of the access switches. On top of this, the solution must be transparent to the rest of the network. In other words, the infrastructure devices themselves, such as the access or aggregation switches must not have any special configuration to make it work.

What are the Necessary Components?

In reality, the solution could have been provided with less than the ten physical servers that were used. Ten were used so that traffic to them could be scaled to some degree (a single physical server emulating 2000 real servers and handling file requests for each of them would quickly be overwhelmed.) The ten servers that were used matched the following specs:

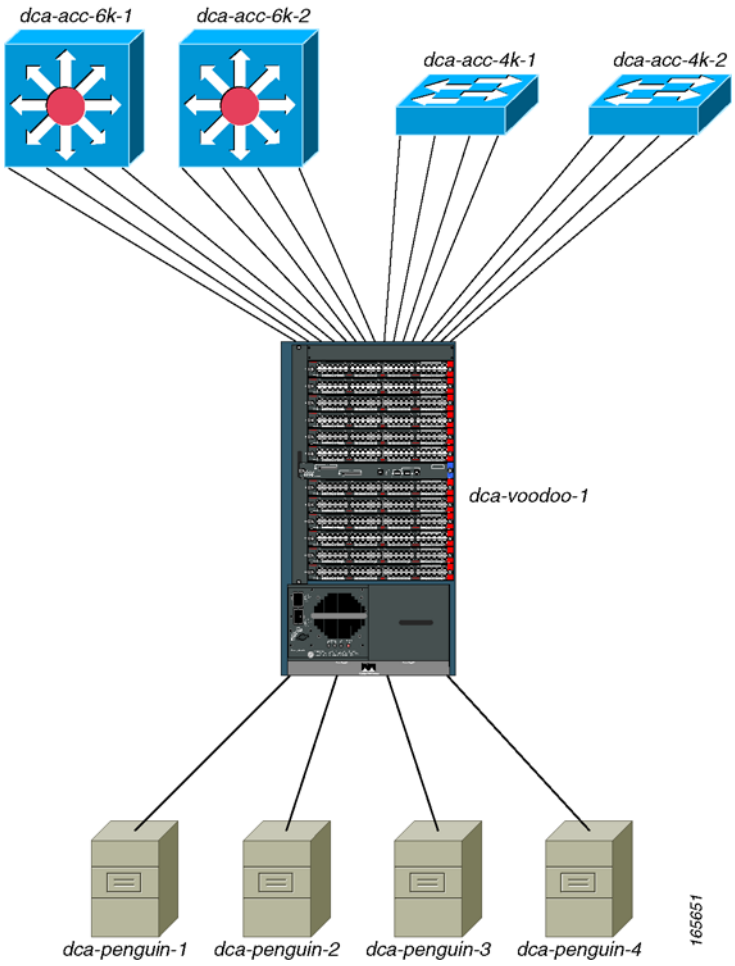
- Dual AMD Opteron processors @ 2600 MHz, w/1 GB onboard cache
- 4 GB DDR RAM
- Red Hat Enterprise Linux

The Catalyst 6513 switch used in the solution SAT between the access switches and the ten servers. It uses a Supervisor 720 as the switching engine and is fully populated with a variety of 48-port line cards (WS-X6148A-GE-TX, WS-X6348-RJ-45 and WS-X6548-RJ-45). The 12 line cards provided a total of 576 ports of Fast Ethernet or Gigabit Ethernet density. In addition to this Catalyst 6513, a second Catalyst 6500 was used to provide connectivity for some of the ten servers (this will be discussed in more detail below).

The Catalyst 6513 that is used to supply connections to the access switches is deployed in the manner described below.

In Figure H-1, the dca-voodoo-1 is fully populated with 48-port linecards, giving it 576 Fast Ethernet and Gigabit Ethernet ports. Four of these ports are used to connect to the four Linux servers via 802.1q trunks. That leaves 572 ports to connect to the Access Layer switches. This is divided nicely among each of the four Linux servers so that each Linux server emulates 143 servers. The Access Layer switch dca-acc-6k-1, a Catalyst 6509 with six WS-X6748-GE-TX linecards, has each of its 288 ports connected into dca-voodoo-1. The top-of-rack Access Layer switch dca-acc-4k-1, a Catalyst 4948, has 47 of its ports connected to dca-voodoo-1 (one port is reserved for management). The remainder of the connections available from dca-voodoo-1 is distributed proportionally between dca-acc-6k-2 and dca-acc-4k-2.

Figure H-1 Catalyst 6513 Used in Voodoo Solution

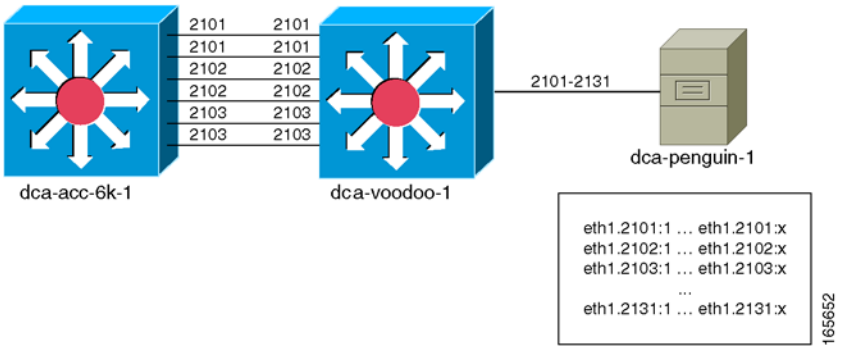


What Features are Used to Make Voodoo Work?

In the DCAP topology, there are only 31 VLANs configured in the Layer 2 domain. These are VLANs 2101-2131. The challenge of using a single physical server to emulate 143 individual hosts cannot be solved by using the same VLAN subinterfaces on the physical servers. In other words, simply configuring eth1.2101 through eth1.2131 on each of the Linux servers would not work for several reasons. First, using only 802.1q subinterfaces would only allow for a maximum of 31 emulated hosts per Linux box. Second, even if virtual interfaces were configured per 802.1q subinterface (eth1.2101:1, eth1.2101:2 and so on) to allow for more than one host per VLAN, nothing is gained towards the goal of having traffic pass across each of the physical links between the Voodoo device and the access switches.

Figure H-2 illustrates this problem. In this diagram, the access switch dca-acc-6k-1 has multiple links on the same VLAN, which reflects the real-world scenario of multiple server hosts being on the same VLAN. The Linux server is configured with 802.1q subinterfaces which, in turn, have multiple virtual interfaces. While traffic could be exchanged between a client and any of the servers emulated by the virtual interfaces on dca-penguin-1, there is no telling what path that traffic will take. An HTTP GET request could pass out of the top link on the access switch on its way to the server, but there's no way to guarantee that the response from the server to the client will use the same path.

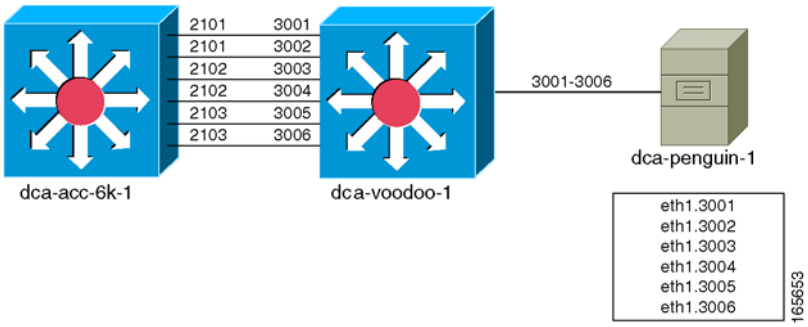
Figure H-2 Limitation of Voodoo with Virtual Interfaces



One other problem with this method is that it does not allow for unique MAC addresses to be configured on a per-virtual interface basis. In the Red Hat Enterprise Linux distribution, unique MAC addresses can be configured for each 802.1q subinterface, but that same MAC is shared with any virtual interfaces configured on that subinterface. Having the ability to configure unique MAC addresses for each of the emulated hosts helps in reflecting the real-world traffic flows.

The Voodoo solution solves the above problems by using a different set of VLANs on the Linux server than are in the DCAP topology Layer 2 domain. On the dca-voodoo-1 side, as illustrated in Figure H-3, each port connecting to an access switch belongs to a unique VLAN. On the Linux server, an 802.1q subinterface is configured for each of the VLANs on the Voodoo device. The important point here is that these Voodoo-only VLANs are only known to the Voodoo device and the Linux server; the actual topology switching infrastructure still only knows about the VLANs in its Layer 2 domain.

Figure H-3 Voodoo Solution Using a Dedicated 802.1q Subinterface for Each Emulated Server



The 802.1q subinterfaces on the Linux server may belong to similar subnets, depending on the DCAP topology VLAN the dca-voodoo-1 port maps to. For example, in Figure H-3, both VLANs 3001 and 3002 on dca-voodoo-1 map to VLAN 2101 on dca-acc-6k-1 and, therefore, are configured with IP addresses in the same subnet. The same holds true for VLANs 3003 and 3004 on the Voodoo device, which both map to VLAN 2102 on the access switch, and for VLANs 3005 and 3006, which map to VLAN 2103.

Thus, there is an allowance for unique MAC addresses to be assigned to each “individual host” emulated by the Linux server. The problem of non-deterministic return paths for emulated hosts belonging to the same subnet has also apparently been solved. Unfortunately, another roadblock is sprung, again stemming from the fact that multiple subinterfaces are sharing a common IP subnet on the Linux server.

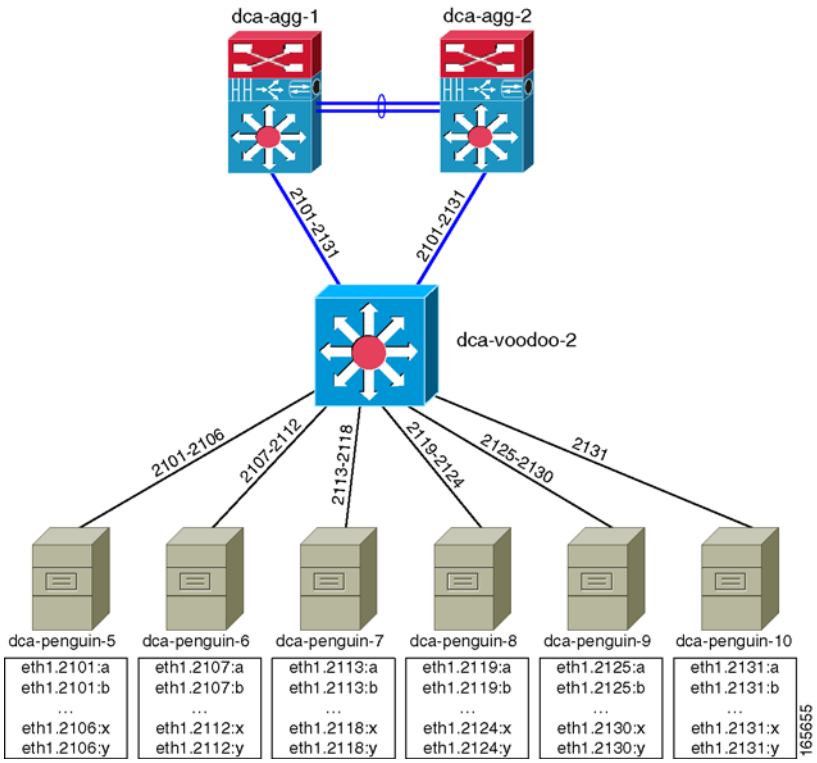
The new problem arises with the usage of the ARP protocol to resolve the MAC address of the default gateway for the emulated servers. It is a given that the 802.1q subinterfaces that share a similar IP subnet also share a common default gateway. So when the Linux box ARPs to resolve the gateway’s IP address, dca-voodoo-1 does not know which port to send the ARP request out. The two ports belonging to VLAN 3001 and 3002 are both set up to carry traffic on the same IP subnet, so when dca-voodoo-1 receives that ARP request, it could choose either of the two ports. (In testing, a single port was chosen for each of the IP subnets.) When the access switch, dca-acc-6k-1, receives the ARP request, it populates it’s MAC table with the MAC address of the Linux subinterface, mapping it to whatever port the ARP request was received on. When traffic flows between client and server, dca-acc-6k-1 sends all downstream traffic through a single port.

To get around this final obstacle, the source routing feature was employed on the Linux server. Using source routing, the Linux box now looks at the source IP address of the packet and sends it out the appropriate 802.1q subinterface. So even though the subinterfaces eth1.3001 and eth1.3002 share a common IP subnet, because the response is coming from one or the other, the proper path will be followed through the Voodoo device. Since the proper path is followed through the Voodoo device, the access switch’s MAC table is populated appropriately. Finally, traffic can deterministically traverse each link in the access layer, making possible a close-to-real-world simulation of multiple server hosts in the datacenter using a single Linux server.

The Voodoo Solution in Full Scale

All that is left to do is increase the scale so that a Catalyst 6509 and Catalyst 4948 can be fully populated, and then some. Figure H-4 shows how that is done in the DCAP topology.

Figure H-4 The full Scale of the Voodoo Solution in the DCAP Test Topology



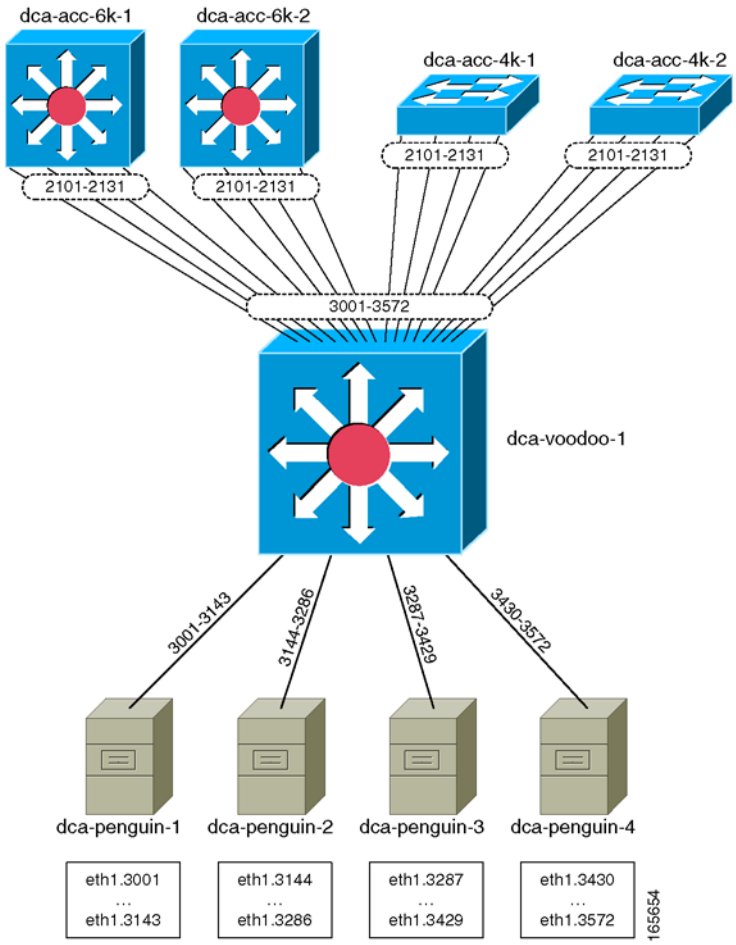
Again, a Catalyst 6513 fully populated with 48-port line cards is used as the Voodoo device, which yields 572 ports for physical links in the access layer. As will be noted below, each Linux server has a limitation on the number of source routes that can be configured (252), so at least 3 Linux servers were needed to fully utilize the capacity of the Catalyst 6513. The total number of Linux servers that were used to emulate these 572 hosts was taken to 4, both for the even divisibility and allocation of subinterfaces, and for the benefits in regards to scaling (4 servers can handle more load than 3).

Each of the Linux servers was configured with 143 802.1q subinterfaces spanning all of the 31 IP subnets used in the DCAP test topology (VLANs 2101-2131). This allowed for each of the four access switches to carry traffic for all subnets as well.

Before details of the configuration of this solution are revealed, what about the other 1428 real servers that were going to be functional in the DCAP topology? Having one fully populated access switch from both the Catalyst 6500 and Catalyst 4900 families was enough, from a testing perspective. While it would have been feasible to scale the Voodoo solution to 2000 real servers, the effort would have been superfluous. One area that was still in need of improved coverage, though, was the scaling of the Aggregation Layer, with regards to 10-Gigabit Ethernet density from the Aggregation Layer to the Access Layer.

Figure H-5 shows how the balance of 1472 real servers was emulated in the DCAP topology. It was through the use of six additional Linux boxes, all connected to the Aggregation Layer through a single Catalyst 6500.

Figure H-5 The 1472 Remaining Servers are Emulated Using Six Additional Linux Hosts



For these remaining servers, virtual interfaces on the Linux hosts were used. Also, unlike the actual Voodoo solution described earlier, each of the Linux hosts here were only configured with a subset of the possible IP subnets, using 802.1q subinterfaces that mapped directly to the VLANs in the Layer 2 domain. Since all of the emulated servers would communicate through a single Catalyst 6500 (dca-voodoo-2), and only one link into the Aggregation Layer would be used at any given time, it is not necessary to use a Voodoo-type setup to force the traffic paths. (The naming of this single Catalyst 6500 dca-voodoo-2 is coincidental; the actual Voodoo solution is not used here.)

Configuration Details

The ports connecting the Voodoo device dca-voodoo-1 to the Access Layer switches were configured as access ports, and assigned to the appropriate VLAN. For example:

```
!
interface GigabitEthernet1/3
 switchport
 switchport access vlan 3001
 switchport mode access
 no ip address
 no cdp enable
 spanning-tree bpdufilter enable
```

```
!
```

The ports connecting dca-voodoo-1 to the four Linux servers were configured as 802.1q trunks, carrying the appropriate VLANs for the respective Linux server.

```
!
interface GigabitEthernet1/1
 description Penguin-1 Eth1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 3001-3143,3600
 switchport mode trunk
 no ip address
 no cdp enable
!
interface GigabitEthernet1/2
 description Penguin-2 Eth1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 3144-3286
 switchport mode trunk
 no ip address
 no cdp enable
!
interface GigabitEthernet3/1
 description Penguin-3 Eth1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 3287-3429
 switchport mode trunk
 no ip address
 no cdp enable
!
interface GigabitEthernet3/2
 description Penguin-4 Eth1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 3430-3572
 switchport mode trunk
 no ip address
 no cdp enable
end
```

Other than that, there is nothing special about the dca-voodoo-1 configuration; it is a simple Layer 2 device.

There were several steps necessary for configuring the 802.1q subinterfaces on each of the Linux servers.

```
# Enable 802.1q trunking
dca-penguin-1$ modprobe 8021q
# Configure the 802.1q subinterfaces on device eth1
dca-penguin-1$ vconfig add eth1 3001
dca-penguin-1$ vconfig add eth1 3002
dca-penguin-1$ vconfig add eth1 3003
...
dca-penguin-1$ vconfig add eth1 3143
# Configure each of the 802.1q subinterfaces with IP and MAC addresses
ifconfig eth1.3001 hw ether 00:00:01:01:30:01 101.1.1.11/24
ifconfig eth1.3002 hw ether 00:00:01:01:30:02 101.1.1.12/24
ifconfig eth1.3003 hw ether 00:00:01:01:30:03 101.1.1.13/24
...
ifconfig eth1.3143 hw ether 00:00:01:01:31:43 101.1.31.14/24
```

Enabling the source routing was also a multi-step process.

The very first thing to do in order to use source routing is to delete the default route entries on the Linux server. Be sure to have an explicit route defined for the management access before doing this.

```
dca-penguin-1$ route del default
```

Each routing entry must first be defined with a name in the file /etc/iproute2/rt_tables. Each entry name is indexed with a line number. The only valid values for line numbers are 1-252.

```
dca-penguin-1$ more /etc/iproute2/rt_tables
#
# reserved values
#
#255    local
#254    main
#253    default
#0      unspec
#
# local
#
#1      inr.ruhep
101     VL3001
102     VL3002
103     VL3003
...
242     VL3142
243     VL3143
```

Next, an IP rule must be added indicating that packets sourced from a particular IP address must use a specific table to be routed.

```
dca-penguin-1$ ip rule add from 101.1.1.11 table VL3001
dca-penguin-1$ ip rule add from 101.1.1.12 table VL3002
dca-penguin-1$ ip rule add from 101.1.1.13 table VL3003
...
dca-penguin-1$ ip rule add from 101.1.31.14 table VL3143
```

The only thing that remains is to tell the Linux box to send any traffic using a certain table out a specified interface.

```
dca-penguin-1$ ip route add default via 101.1.1.1 dev eth1.3001 table VL3001
dca-penguin-1$ ip route add default via 101.1.1.1 dev eth1.3002 table VL3002
dca-penguin-1$ ip route add default via 101.1.1.1 dev eth1.3003 table VL3003
...
```

dca-penguin-1$ **ip route add default via 101.1.1.1 dev eth1.3143 table VL3143**

APPENDIX I

Bill of Materials and Power Draw

This appendix provides a bill of materials for Cisco equipment tested in Cisco DCAP 4.0. It is broken down by device, and includes real power draw information taken with the device in idle state.

Table I-1 Cisco DCAP 4.0 Bill of Materials and Power Draw

Device Name	Hardware List	Power Draw (watts)
dca-core-1	WS-C6506-E WS-CAC-3000W WS-C6506-E-FAN WS-SUP720-3B WS-X6704-10GE WS-X6748-GE-TX WS-F6700-DFC3BXL	856
dca-core-2	WS-C6506-E WS-CAC-2500W WS-C6506-E-FAN WS-SUP720-3B WS-X6704-10GE	696

Table I-1 Cisco DCAP 4.0 Bill of Materials and Power Draw (continued)

Device Name	Hardware List	Power Draw (watts)
dca-agg-1	WS-C6513 WS-CAC-6000W WS-C6K-13SLT-FAN2 WS-SUP720-3B WS-SVC-FWM-1 ACE10-6500-K9 WS-SVC-IDSM-2 WS-SVC-IDSUPG WS-SVC-NAM-2 WS-X6708-10GE (4) WS-X6748-GE-TX WS-F6700-DFC3CXL (2) WS-F6700-DFC3C (2)	2536
dca-agg-2	WS-C6513 WS-CAC-6000W WS-C6K-13SLT-FAN2 WS-SUP720-3B WS-SVC-FWM-1 ACE10-6500-K9 WS-SVC-IDSM-2 WS-SVC-IDSUPG WS-SVC-NAM-2 WS-X6704-10GE (4) WS-X6748-GE-TX WS-F6700-DFC3B (4)	2496
dca-acc-6k-1	WS-C6509-E WS-CAC-6000W WS-C6509-E-FAN WS-SUP720-3B (2) WS-X6704-10GE WS-X6748-GE-TX (6)	2273

Table I-1 Cisco DCAP 4.0 Bill of Materials and Power Draw (continued)

Device Name	Hardware List	Power Draw (watts)
dca-acc-6k-2	WS-C6509-E WS-CAC-6000W WS-C6509-E-FAN WS-SUP720-3B (2) WS-X6704-10GE WS-X6748-GE-TX (6)	2273
dca-acc-4k-1	WS-C4948-10GE	240
dca-acc-4k-2	WS-C4948-10GE	240
dcb-core-1	WS-C6506-E WS-CAC-3000W (2) WS-C6506-E-FAN WS-SUP720-3BXL WS-X6704-10GE WS-X6748-GE-TX WS-F6700-DFC3BXL	844
dcb-core-2	WS-C6506-E WS-CAC-2500W (2) WS-C6506-E-FAN WS-SUP720-3BXL WS-X6704-10GE WS-F6700-DFC3BXL	659
dcb-agg-1	WS-C6509-E WS-CAC-6000W (2) WS-C6509-E-FAN WS-SUP720-3BXL WS-X6708-10GE (7) WS-X6748-GE-TX WS-F6700-DFC3C (7)	3009
dcb-agg-2	WS-C6509-E WS-CAC-6000W WS-C6509-E-FAN WS-SUP720-3BXL WS-X6704-10GE (7) WS-X6748-GE-TX WS-F6700-DFC3BXL (7)	2522

Table I-1 Cisco DCAP 4.0 Bill of Materials and Power Draw (continued)

Device Name	Hardware List	Power Draw (watts)
dcb-ss-1	WS-C6509-E WS-CAC-3000W WS-C6509-E-FAN WS-SUP720-3B WS-SVC-FWM-1 WS-X6066-SLB-APC WS-SVC-SSL-1 (2) WS-SVC-IDSM-2 WS-SVC-IDSUPG WS-SVC-NAM-2 WS-X6704-10GE	1366
dcb-ss-2	WS-C6509-E WS-CAC-3000W WS-C6509-E-FAN WS-SUP720-3B WS-SVC-FWM-1 WS-X6066-SLB-APC WS-SVC-SSL-1 (2) WS-SVC-IDSM-2 WS-SVC-IDSUPG WS-SVC-NAM-2 WS-X6704-10GE WS-F6700-DFC3BXL	1406
dcb-acc-6k-1	WS-C6509-E WS-CAC-6000W WS-SUP720-3BXL (2) WS-X6704-10GE WS-X6748-GE-TX (6) WS-F6700-DFC3BXL	2273
dcb-acc-6k-2	WS-C6509-E WS-CAC-6000W WS-SUP720-3BXL (2) WS-X6704-10GE WS-X6748-GE-TX (6)	2273
dcb-acc-4k-1	WS-C4948-10GE	240

Table I-1 Cisco DCAP 4.0 Bill of Materials and Power Draw (continued)

Device Name	Hardware List	Power Draw (watts)
dcb-acc-4k-2	WS-C4948-10GE	240
dcap-m9513-north1	DS-C9513 DS-X9032-SSM DS-X9302-14K9 DS-X9530-SF2-K9 (2) DS-X9112 (2)	1203
dcap-m9513-south1	DS-C9513 DS-X9032-SSM DX-X9302-14K9 DS-X9530-SF2-K9 (2) DS-X9112 (2)	1203
dcap-m9513-core-a1	DS-C9513 DS-X9112 (2) DS-X9530-SF2-K9 (2) DS-X9124 (2)	1075
dcap-m9513-core-a2	DS-C9513 DS-X9112 (2) DS-X9530-SF2-K9 (2) DS-X9124 (2)	1075
dcap-m9513-core-b1	DS-C9513 DS-X9112 (2) DS-X9530-SF2-K9 (2) DS-X9124 (2)	1075
dcap-m9513-core-b2	DS-C9513 DS-X9112 (2) DS-X9530-SF2-K9 (2) DS-X9124 (2)	1075
dcap-m9513-edge-a1	DS-C9513 DS-X9148 DS-X9112 (2) DS-X9530-SF2-K9 (2)	1035
dcap-m9513-edge-a2	DS-C9513 DS-X9148 DS-X9112 (2) DS-X9530-SF2-K9 (2)	1035

Table I-1 Cisco DCAP 4.0 Bill of Materials and Power Draw (continued)

Device Name	Hardware List	Power Draw (watts)
dcap-m9513-edge-b1	DS-C9513 DS-X9148 DS-X9112 (2) DS-X9530-SF2-K9 (2)	1035
dcap-m9513-edge-b2	DS-C9513 DS-X9148 DS-X9112 (2) DS-X9530-SF2-K9 (2)	1035
dcap-m9513-north2	DS-C9513 DS-X9032-SSM DS-X9302-14K9 DS-X9530-SF2-K9 (2) DS-X9112 (2)	1203
dcap-m9513-south2	DS-C9513 DS-X9032-SSM DS-X9302-14K9 DS-X9530-SF2-K9 (2) DS-X9112 (2)	1203
dcap-m9513-core-c1	DS-C9513 DS-X9112 (2) DS-X9530-SF2-K9 (2) DS-X9124 (2)	1075
dcap-m9513-core-c2	DS-C9513 DS-X9112 (2) DS-X9530-SF2-K9 (2) DS-X9124 (2)	1075
dcap-m9509-core-d1	DS-C9509 DS-X9112 (2) DS-X9530-SF2-K9 (2) DS-X9124 (2)	872
dcap-m9509-core-d2	DS-C9509 DS-X9112 (2) DS-X9530-SF2-K9 (2) DS-X9124 (2)	872

Table I-1 Cisco DCAP 4.0 Bill of Materials and Power Draw (continued)

Device Name	Hardware List	Power Draw (watts)
dcap-m9513-edge-c1	DS-C9513 DS-X9148 DS-X9112 (2) DS-X9530-SF2-K9 (2)	1035
dcap-m9513-edge-c2	DS-C9513 DS-X9148 DS-X9112 (2) DS-X9530-SF2-K9 (2)	1035
dcap-m9509-edge-d1	DS-C9509 DS-X9148 DS-X9112 (2) DS-X9530-SF2-K9 (2)	832
dcap-m9509-edge-d2	DS-C9509 DS-X9148 DS-X9112 (2) DS-X9530-SF2-K9 (2)	832
dca-gss-1	GSS-4492R-K9	240
dca-gss-2	GSS-4492R-K9	240
dcb-gss-1	GSS-4492R-K9	240
dcb-gss-2	GSS-4492R-K9	240
dca-wae-7326-1	WAE-7326-K9	600
dca-wae-7326-2	WAE-7326-K9	600
dca-wae-512-cm	WAE-512-K9	120
dcb-wae-7326-1	WAE-7326-K9	600
dcb-wae-7326-2	WAE-7326-K9	600
dcb-wae-512-cm	WAE-512-K9	120
wae-branch1-512-1	WAE-512-K9	120
wae-branch1-612-1	WAE-612-K9	360
wae-3845-branch1	CISCO3845	120
wae-2821-branch2	CISCO2821 NMA-WAE-502-K9	120
wae-branch2-512-1	WAE-512-K9	120
wae-2811-branch3	CISCO2811 NMA-WAE-502-K9	120

APPENDIX J

DCAP 4.0 Resources

DCAP testing relies on many sources to guide the successful deployment of the various data center features and technologies. Cisco Solution Reference Network Designs (SRNDs) are used where possible as a baseline for test bed design. Other sources for design guidance include Cisco configuration guides as well as vendor white papers and implementation guides. Below is a collection of resources that were used in the design and implementation of the DCAP 4.0 topology and testing.

Cisco Resources

Data Center Infrastructure Design Guide 2.1

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns107/c649/ccmigration_09186a008073377d.pdf

Data Center Infrastructure Design Guide 2.1 Readme File

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns107/c133/ccmigration_09186a0080733855.pdf

Data Center Infrastructure Design Guide 2.1 Release Notes

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns107/c133/ccmigration_09186a00807337fc.pdf

Server Farm Security in the Business Ready Data Center Architecture v2.1

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns376/c649/ccmigration_09186a008078e021.pdf

Enterprise Data Center Wide Area Application Services

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns377/c649/ccmigration_09186a008081c7da.pdf

Data Center Blade Server Integration Guide

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns304/c649/ccmigration_09186a00807ed7e1.pdf

Integrating Oracle E-Business Suite 11i in the Cisco Data Center

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns50/c649/ccmigration_09186a00807688ce.pdf

Cisco SAN Interoperability Matrix

http://now.netapp.com/NOW/knowledge/docs/san/fcp_iscsi_config/fcp_switch.shtml

Cisco MDS 9000 Family Fabric Manager Configuration Guide, Release 3.x

http://www.cisco.com/en/US/products/ps5989/products_configuration_guide_book09186a0080667aa0.html

Cisco MDS 9000 Family CLI Configuration Guide, Release 3.x

http://www.cisco.com/en/US/products/ps5989/products_configuration_guide_book09186a00806688da.html

Data Center

SAN Extension for Business Continuance

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns516/c649/cdccont_0900aecd8023dd9d.pdf

EMC Resources

EMC Interoperability Matrix

http://www.emc.com/interoperability

Oracle Databases on EMC Symmetrix Storage Systems

http://www.emc.com/techlib/pdf/H2603_oracle_db_emc_symmetrix_stor_sys_wp_ldv.pdf

EMC and Cisco: Building Disaster Recovery and Business Continuance Solutions

http://www.emc.com/partnersalliances/partner_pages/pdf/H1182_emc_cisco_wp_ldv.pdf

Exchange 2003 Recovery—Rapid Local Recovery versus Disaster Recovery

http://www.emc.com/techlib/pdf/H1645_ExchangeRecovery2003_ldv.pdf

ESRP Storage Program EMC Symmetrix DMX-3 4500 SRDF/S (60,000 Users) Storage Solution for Microsoft Exchange Server Replicated Storage

http://www.emc.com/techlib/pdf/CSG1566_esrp_dmx_3_4500_srdf_s_6000_user_wp_ldv.pdf

HP Resources

HP Interoperability Matrix

http://h18006.www1.hp.com/storage/networking/index.html ("Fabric Infrastructure Rules HP StorageWorks SAN Design Reference Guide" for C Series).

HP StorageWorks Continuous Access XP user guide for the XP12000/XP10000/SVS200

http://h20000.www2.hp.com/bc/docs/support/SupportManual/c00801872/c00801872.pdf

HP StorageWorks Continuous Access XP Journal user guide for the XP12000/XP10000/SVS200

http://h20000.www2.hp.com/bc/docs/support/SupportManual/c00942547/c00942547.pdf

HP Best Practices for Replication of Oracle Storage Area Networks White Paper

http://h71028.www7.hp.com/ERC/downloads/4AA0-1650ENW.pdf

HP StorageWorks XP Disk Array Design Considerations for Microsoft Exchange 2003 - White Paper

http://h71028.www7.hp.com/ERC/downloads/5982-7883EN.pdf

Microsoft Resources

How to Move All Exchange Virtual Servers from a Production Exchange 2003 Cluster to a Standby Exchange 2003 Cluster

http://technet.microsoft.com/en-us/library/aa996470.aspx

Exchange Server 2003 Advanced Recovery Strategies

http://www.microsoft.com/technet/prodtechnol/exchange/guides/DROpsGuide/f4d7aa56-abed-4645-b2f8-952191d1c050.mspx

Microsoft Exchange Server 2003 Load Simulator (LoadSim)

http://go.microsoft.com/fwlink/?linkid=27882

Microsoft Exchange Server Jetstress Tool (32 bit)

http://go.microsoft.com/fwlink/?linkid=27883

Network Appliance Resources

NetApp Interoperability Guide

http://now.netapp.com/NOW/knowledge/docs/san/fcp_iscsi_config/fcp_switch.shtml

Using Synchronous SnapMirror® for Disaster Protection with Block-Access Protocols

http://www.netapp.com/library/tr/3324.pdf

Oracle Resources

Oracle Clusterware and Oracle Real Application Clusters Installation Guide

http://download.oracle.com/docs/cd/B19306_01/install.102/b14203.pdf

Oracle Database Installation Guide 10gR2 for Linux x86-64

http://download.oracle.com/docs/cd/B19306_01/install.102/b15667.pdf

Oracle RAC Administration and Deployment Guide

http://download.oracle.com/docs/cd/B19306_01/rac.102/b14197.pdf

Oracle Database 10gR2 ASM Best Practices

http://www.oracle.com/technology/products/database/asm/pdf/asm_10gr2_bestpractices%2009-07.pdf

Oracle Maximum Availability Architecture

http://www.oracle.com/technology/deploy/availability/htdocs/maa.htm

Configuring Oracle Applications Release 11i with 10g RAC and 10g ASM (Note: 312731.1)

http://metalink.oracle.com

Symantec (Veritas) Resources

VERITAS NetBackup (tm) 6.0 Media Manager System Administrator's Guide for UNIX:
http://seer.entsupport.symantec.com/docs/279267.htm

VERITAS NetBackup (tm) 6.0 System Administrator's Guide for UNIX, Volume 1:
http://seer.entsupport.symantec.com/docs/279263.htm

VERITAS NetBackup (tm) 6.0 System Administrator's Guide for UNIX, Volume 2:
http://seer.entsupport.symantec.com/docs/279264.htm

VERITAS NetBackup (tm) 6.0 Troubleshooting Guide for UNIX and Windows:
http://seer.entsupport.symantec.com/docs/279295.htm

APPENDIX **K**

Safe Harbor Technology Releases

The DCAP testing effort often relies on testing performed by other teams, particularly the Safe Harbor team. The determination of which software to run in the various systems in the DCAP topology is made based on Safe Harbor software recommendations. Many of the tests executed in regular Safe Harbor testing are applicable to the DCAP topology and are leveraged for the final DCAP product. While those test results are considered in the final result, they are not reported in this document. Refer to the appropriate technolgy release for the latest technology release testing details.

Table K-1 lists the EDCS document numbers so the reader can locate and review results of relevant Safe Harbor testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested. Safe Harbor is designed to cover critical path areas and augment ongoing regression and systems testing.

Table K-1 Cisco Safe Harbor Technology Releases in EDCS

Platform	Software Version	EDCS Doc. No.
Application Control Engine	3.0(0)A1(6.3)	659550
Content Switching Module	4.2(6)	605555
Firewall Services Module	3.2(4)	664569
Intrusion Detection Services Module 6.0.3	3.2(4)	652756
Native IOS Supervisor 720	12.2(18)SXF12a	660804
Secure Socket Layer Module	2.1(11)	504166
Wide Area Application Services	4.0.13.23	633506

The following summary tables list tests conducted in the latest Safe Harbor technology releases.

Application Control Engine (ACE) 3.0(0)A1(6.3)

The following Safe Harbor Application Control Engine (ACE) Release 3.0(0)A1(6.3)/12.2(18)SXF11 was tested and certified as a conditional recommendation.

Table K-2 summarizes testing executed as part of the Cisco Application Control Engine (ACE) Release 3.0(0)A1(6.3) Safe Harbor initiative. These tables include component tests for each feature or function.

Table K-2 Cisco Safe Harbor Test Results Summary

Test Suites	Feature/Function	Tests	Results
Baseline	Baseline Network Steady State	Baseline Network Steady State	
		Baseline Network Validation	CSCsm64949 CSCsl95565 CSCsm35407 CSCsl89772
Basic Functionality	CLI	Addition—Removal of Rservers (Bridged ACE config)	
		Addition—Removal of Rservers	
	Device Management	ACE XLM Negative	
		SNMP	
		TACACS	
		XML	
	Hardware Reliability	Repeated FT Port Channel Flap	CSCsm39305
		Repeated Link Flap Client Side Interface	
		Repeated Link Flap Client Side Vlan	
		Repeated Link Flap FT Interface	
		Repeated Link Flap Server Side Interface	
		Repeated Link Flap Server Side Vlan	
		Repeated Port-Channel Flap Client Side	
		Repeated Port-Channel Flap Server Side	
	SPAN	Distributed Etherchannel RSPAN	
		Distributed Etherchannel SPAN	CSCsm32580
	Traffic Decisions	Failaction Purge (Bridged ACE config)	
		Failaction Purge	
		ICMP (Bridged ACE config)	
		ICMP	CSCsi60550 CSCsi62321
		Idle Connections (Bridged ACE Config)	CSCsm83119 CSCsm83535
		Idle Timeout	CSCsk38649 CSCsm80600
		Route Health Injection	

Table K-2 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
	Upgrade	Dual ACE Upgrade From Previous Certified ACE Release	CSCsl77474
		Dual Supervisor Upgrade From Previous Certified Native Release	
		Single ACE Upgrade From Previous Certified ACE Release	
		Single Supervisor Upgrade From Previous Certified Native Release	
Health and Redundancy	Backup Serverfarm	Backup Serverfarm	
	Configuration Syncronization	Config Sync Large	CSCsh63341
		Configuration Sync	CSCsh63341 CSCsj68643
	Probes	DNS Probes (Bridged ACE Config)	
		DNS Probe	
		HTTP Probes (Bridged ACE Config)	CSCsm78622
		HTTP Probes	
		Probe Negative	
		SSL Probe (Bridged ACE Config)	CSCsm83326 CSCsm83523
		SSL Probe	CSCsk16083
		Scripted Probes (Bridged ACE Config)	
		Scripted Probes	CSCsi43733
	Redundancy	Multiple Chassis Redundancy (Bridged ACE Config)	
		Multiple Chassis Redundancy	
		Single Chassis Redundancy (Bridged ACE Config)	
		Single Chassis Redundancy	CSCso04799
	Resets	Module Reset	
		Port Reset (Bridged ACE Config)	
		Port Reset	
		VLAN Reset (Bridged ACE Config)	
		VLAN Reset	
	Tracking	Fault Tolerant Tracking (Bridged ACE Config)	
		Fault Tolerant Tracking	

Table K-2 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Load Balancing	Predictors	IP Addr Hash	
		Least Connection Predictor	CSCsi07311
		Maxconn Connection Limiter	CSCsm12883
		Round Robin	
		Server Weight	
		URL Hashing	CSCsm10896
Memory Testing	Leaks	Repeated SSHv1 Logins of Primary ACE Module	
		Repeated SSHv2 Logins of Primary ACE Module	
		Repeated Telnet Logins of Primary ACE Module	
Parser	Parser	Parser ACE via SSH (Bridged ACE Config)	CSCsm08597 CSCsm17268
		Parser ACE via SSH	CSCsm08597 CSCsm17268
		Parser ACE via Telnet (Bridged ACE Config)	CSCsm17268
		Parser ACE via Telnet	CSCsm08597 CSCsm17268
Traffic Handling	FTP	FTP—Active	
		FTP—L7 Inspection	
		FTP—Passive	
	Insert	Cookie Insert	
		Header Insert	
	Maps	Cookie Map	
		Header Map	
		URL Map	
	Miscellaneous	L4 to L7 Policy Change	
		Persistence Rebalance	
		Pipeline	
		Redirect Policy (Bridged ACE Config)	
		Redirect Policy	
		TCP Anomalous Traffic	CSCsm73606
		TCP Normalization	
		TCP Reuse	
		UDP Load Balancing (Bridged ACE Config)	
		UDP Load Balancing	CSCsg80625
		URL Lengths	

Table K-2 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
	SSL	SSL End To End SSL Initiation SSL Termination	
	Sticky	Cookie Sticky Header Sticky IP Netmask Sticky	CSCsm57955
Miscellaneous	Virtualization Duplicate IP	Virtualization Duplicate IP	CSCsi62078
	Virtualization	Virtualization	CSCsd85145

Content Switching Module (CSM) 4.2.6

The following Safe Harbor Content Switching Module (CSM) 4.2.6/12.2(18)SXF6 was tested and certified as a conditional recommendation.

Table K-3 and Table K-4 summarizes testing executed as part of the Cisco Safe Harbor Router Mode Content Switching Module (CSM) Release 4.2.6/12.2(18)SXF6 initiative. These tables include features or functions tested for routed and bridged mode functionality, respectively, and component tests for each feature or function.

Table K-3 Cisco Safe Harbor Routed Mode Test Results Summary

Features/Functions	Tests	Results
CSM Basic Functionality, page 14	Additional Removal of Servers	
	CLI Parser	
	DEC rSPAN CSM Load Balance—Sup 720	CSCek67352
	DEC SPAN CSM Load Balance—Sup 720	
	Failaction Purge	
	Lifetime of Idle Connections	
	Route Health Injection	CSCek67326
	SNMP MIBs Traps	
	XML Negative Testing	
	XML	
Health & Redundancy, page 33	Backup Serverfarm	
	Config Synch	CSCek70457
	Config Synch Large	
	Health Probes	
	Interface Tracking	
	Interswitch Redundancy	
	Intraswitch Redundancy	
	Module Reset	
	Port Reset	
	VLAN Reset	

Table K-3 Cisco Safe Harbor Routed Mode Test Results Summary (continued)

Features/Functions	Tests	Results
Load Balancing Predictors, page 66	IP Address Hash	
	Least Connection	
	MaxConn	CSCek68161
	Predictor Round Robin	
	SASP Bindid	
	SASP Invalid Message Length	
	SASP Load Balancing	CSCek68456
	SASP Protocol Version	
	SASP Registration and Member State	
	SASP Scaling	
	SASP Weights	
	SASP Wrong Predictor	
	Server Weighting	
	URL Hash	
Traffic Handling, page 101	Anomalous TCP	
	Cookies Insert	
	Cookies Maps	
	Cookies Sticky	
	FTP Active	
	FTP Passive	
	Header Insert	CSCek71183
	Header Maps	
	Header Sticky	CSCek69968
	Load Balance Non TCP	
	Netmask Sticky	
	Non Secure Routed Mode	
	Persistence Rebalance	
	Ping Handling	CSCek70084, CSCek70088
	Policy Ordering	
	Redirect Policy	
	SSL Sticky	
	URL Lengths	
	URL Maps	
	VIP Dependency	CSCek68612, CSCek68627

Table K-4 Cisco Safe Harbor Bridged Mode Test Results Summary

Features/Functions	Tests	Results
CSM Basic Functionality, page 152	Additional Removal of Servers—Bridged	
	Failaction Purge—Bridged	
	Lifetime of Idle Connections—Bridged	
Health & Redundancy, page 161	Health Probes—Bridged	
	Interface Tracking	
	Interswitch Redundancy	CSCsi01910, CSCek72445
	Intraswitch Redundancy	CSCek72566
	Port Reset—Bridged	
	VLAN Reset—Bridged	
Traffic Handling, page 185	Load Balance Non TCP	
	Ping Handling—Bridged	CSCek59900
	Redirect Policy—Bridged	
	VIP Dependency—Bridges	CSCek71982

Firewall Services Module (FWSM) 3.2.4

The following Safe Harbor Firewall Services Module (FWSM) Release 3.2.4/12.2(18)SXF11 was tested and certified as a conditional recommendation.

Table K-5 summarizes testing executed as part of the Cisco Safe Harbor Firewall Services Module (FWSM) 3.2.4/12.2(18)SXF11 initiative. This table includes the technology tested, the feature or function tested, and the component tests for each feature or function.

Table K-5 Cisco Safe Harbor Test Results Summary

Test Suites	Feature/Function	Tests	Results
AAA	AAA Management Traffic, page 17	RADIUS Fallback for Management Traffic	
		RADIUS Management Traffic	
		TACACS Fallback for Management Traffic	CSCsk48291
AAA	AAA Network Traffic, page 26	HTTPS Cut-Through Proxy	
		Multiple AAA Servers	CSCsh98336
		RADIUS Authentication Stress	
		RADIUS High to Low and Low to High Security	CSCsh98336
		RADIUS Network Authorization	CSCsh98336
		RADIUS on Same Security Interface	CSCsh98336
		TACACS High to Low Security	
		TACACS One Client using Multiple IPs	
		Virtual Server	
ACL	Multi Routed ACL, page 56	Modify Access List	
		Multi Routed ACL Logging	
		Multi Routed ACL Object Group Manual Commit	
		Multi Routed ACL Object Group	
ACL	Single Routed ACL, page 68	Single Routed 1K Inbound ACL	CSCsh66769
		Single Routed 1K Outbound ACL Manual Commit	CSCsh66769
		Single Routed 500 Outbound ACL	CSCsh66769
		Single Routed 50k ACL Auto Commit	
		Single Routed 50k ACL Manual Commit	
		Single Routed 5k AAA ACL Stress	
		Single Routed ACL Object Group	
		Timed ACL Verification	

Table K-5 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
ACL	Single Transparent ACL, page 88	Single Transparent 1K Inbound ACL Manual Commit	
		Single Transparent 1K Inbound ACL	CSCsh66769
		Single Transparent 1K Outbound ACL	CSCsh66769
		Single Transparent 50K ACL Stress	
		Single Transparent 50K Manual Commit Stress	
		Single Transparent ACL Object Group	
Advanced Conn Features	Connection Timeout, page 103	Connection Timeout	
		Connection Timeout	
		Connection Timeout	
		Connection Timeout	
Advanced Conn Features	TCP State Bypass, page 116	TCP State Bypass	CSCsm53984
Application Layer Protocol Inspection	DNS Inspection, page 120	DNS Fixup Stress	
		Inspect DNS NAT	
		Max Length DNS Reply	CSCsk01370
Application Layer Protocol Inspection	FTP Inspection, page 130	FTP Fixup—Active FTP	
		FTP Fixup—PASV FTP	CSCsh98336
		FTP Fixup—Stress	
Application Layer Protocol Inspection	ICMP Inspection, page 139	ICMP Error Inspection	
		ICMP Fixup	
Application Layer Protocol Inspection	SIP Inspection, page 145	SIP Fixup Stress	
		SIP Fixup UDP	
Application Layer Protocol Inspection	SMTP Inspection, page 151	SMTP Fixup	
		SMTP Stress	
FWSM Management	Capture, page 159	Capture FTP	
		Capture HTTP	
		Capture ICMP	CSCsf16564
FWSM Management	Cisco Security Manager, page 167	Cisco Security Manager Basic Configuration	
		Limit Connection Rate with Cisco Security Manager	
		Modify Acess-list with Cisco Security Manager	CSCso28857
		New Context with Cisco Security Manager	

Table K-5 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
FWSM Management	Configuration, page 179	Multi Routed Mode Create Multiple Contexts	
		Multi Transparent Mode Create Multiple Contexts	
		Single Routed Change Interface IP Address	
		Single Routed Change Interface Name and IP Address	
		Single Routed Change Interface Name	
		Single Routed Remove and Reconfigure Interface	
		Single Transparent Change Interface IP Address	
		Single Transparent Change Interface Name and IP Address	
		Single Transparent Change Interface Name	
		Single Transparent Remove and Reconfigure Interface	
FWSM Management	Parser, page 200	Multi Routed Mode Parser SSH	CSCsl54834
		Multi Routed Parser	CSCsl54834
		Multi Transparent Mode Parser SSH	CSCsl54834
		Multi Transparent Parser	CSCsl54834
		Single Routed Parser SSH	CSCsl54834
		Single Routed Parser	CSCsl54834
		Single Transparent Mode Parser SSH	CSCsl54834
		Single Transparent Parser	CSCsl54834
FWSM Management	Resource Management, page 216	Connection Rate Across Contexts	
		Fixup Rate Across Contexts	
		Multi Routed Total Xlate Cross Contexts	
		Syslog Rate Across Context	
		Total Connections Across Contexts	
		Total SSH Sessions Across Contexts	
FWSM Management	SNMP, page 234	Multi Routed SNMP Walk	
		Multi Transparent SNMP Walk	
		Single Routed SNMP Walk	
		Single Transparent SNMP Walk	
FWSM Management	Syslog, page 243	Syslog Functionality	
		Syslog Performance	
		Syslog Standby	
Multicast	Multicast, page 252	IGMP V2 Functionality	
		IGMP V2 Stress	
		Low to High Security Multicast Traffic	
		Multicast Stress	

Table K-5 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
NAT	Connection Limits and TCP Interception, page 266	SYN Cookie Active FTP Attack SYN Cookie HTTP Attack SYN Cookie Passive FTP Attack SYN Cookie TCP SYN Attack TCP and UDP Connection Rate	
NAT	NAT Between Same Security Level Interfaces, page 280	NAT Between Same Security Interface No NAT Between Same Security	
NAT	NAT High Security Interface to Low Security Interface, page 286	Bidirectional NAT Dynamic Identity NAT Dynamic NAT Plus PAT Dynamic NAT Dynamic PAT NAT Exemption NAT Exemption NAT Outside Policy NAT STFW Dynamic NAT STFW Dynamic PAT STFW Policy NAT Static Identity NAT	CSCsh66769
NAT	NAT Low Security Interface to High Security Interface, page 322	Static NAT Static PAT Static Policy PAT	
NAT	Xlate Bypass, page 330	Multi Routed Mode Xlate Bypass Single Routed Mode Xlate Bypass Single Transparent Mode Xlate Bypass	CSCsh66769
OSPF	OSPF, page 340	OSPF Distribution OSPF Interface Parameters	

Table K-5 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Redundancy	Multi Routed Mode Redundancy, page 346	Multi Routed Config Sync Between Active Standby Multi Routed Data Links Failover Multi Routed HTTP Traffic Failover Multi Routed Long Lived Traffic Failover Multi Routed Mode Active Active Failover Multi Routed Stateful Traffic Failover Multi Routed Stateless Traffic Failover Multi Routed Suspend Config Sync	
Redundancy	Multi Transparent Mode Redundancy, page 373	Multi Transparent Link to Access Switch Failover Multi Transparent Long Lived Traffic Failover Multi Transparent Mode Active Active Failover Multi Transparent Stateful Traffic Failover Multi Transparent Stateless Traffic Failover Multi Transparent Suspend Config Sync Failover Multi Transparent Switch Failover	
Redundancy	Single Routed Mode Redundancy, page 397	Single Routed Long Lived Traffic Failover Single Routed Mode Failover with Preempt Single Routed Stateful Links Failover Single Routed Stateful Traffic Failover Single Routed Stateless Traffic Failover Single Routed Suspend Config Sync Failover Single Routed Switch Failover	
Redundancy	Single Transparent Mode Redundancy, page 421	Single Transparent Data Links Failover Single Transparent Mode Failover with Preempt Single Transparent Multicast Traffic Failover Single Transparent Switch Failover	

Intrusion Detection Services Module (IDSM) Release 6.0.3

The following Safe Harbor Intrusion Detection Services Module (IDSM) Release 6.0.3/12.2(18)SXF9 was tested and certified as a conditional recommendation.

Table K-6 summarizes testing executed as part of the Cisco Safe Harbor Intrusion Detection Services Module (IDSM) Release 6.0.3/12.2(18)SXF9 Safe Harbor initiative. This table include component tests for each feature or function.

Table K-6 Cisco Safe Harbor Test Results Summary

Feature/Function	Tests	Results
Detection, page 7	Anomaly Detection (AD)	
	FTP Detection	
	HTTP Detection	
	ICMP Detection	
	IP detection	
	SNMP Detection	
	TCP Detection	
	UDP Detection	
Evasion, page 25	Fragmentation	
	HTTP Obfuscation	
Event Action, page 30	Blocking	
	Logging	
	Passive OS Fingerprinting	CSCsm04146
	SNMP Traps	
	TCP Reset	
General, page 43	External Product Interface	
	IDSM2 Boot Into Maintenace Partition	
	NTP	
Load Testing, page 50	300 Mbps Background Traffic	
	480 Mbps Background Traffic	
Modes, page 55	Inline Mode	
	Inline VLAN Pair with EtherChannel Load Balancing	
	Inline VLAN Pair	
	Inline with EtherChannel Load Balancing	
	Promiscous with EtherChannel Load Balancing	
	Promiscuous Mode	
	Virtualization	
Upgrades, page 74	IDSM2 Upgrade	
	Signature Update	

Native (Classic) IOS 12.2(18)SXF12a

The following Safe Harbor Native (Classic) IOS Release 12.2(18)SXF12a was tested and certified as a conditional recommendation.

Table K-7 summarizes tests executed as part of the Cisco Safe Harbor Campus IOS Release 12.2(18)SXF12a initiative. This table includes the feature or function tested, the section that describes the feature set the feature or function belongs to, and the component tests for each feature or function.

Table K-7 Cisco Safe Harbor Test Results Summary

Test Suites	Feature/Function	Tests	Results
FHRP	GLBP, page 3-34	GLBP Basic Function	
FHRP	HSRP, page 3-38	Distributed GE Module Failover Sup 720	
		Failover with Default Timers Sup 720	
		Failover with Default Timers Sup22	
		Failover with Fast Timers Sup22	
		Failover with Fast Timers Sup720	
		Maximum Group Limit Sup22	
		Maximum Group Limit Sup720	
		Recovery from System Failure Sup22	
		Recovery from System Failure Sup720	
		Traffic Impact on CPU Sup22	
		Traffic Impact on CPU Sup720	
Layer2	Distributed Etherchannel, page 3-67	DEC Traffic Should Not Flood to all Linecards Sup22	CSCsm65726
		DEC Traffic Should Not Flood to all Linecards Sup720	
		L2 DEC Flooding Due to Periodic Purge Sup22	
		L2 DEC Lost PI E Due to Timeout Sup22	
		L2 DEC Lost PI E Due to Timeout Sup720	
		L2 DEC MAC Out of Band Stress Sup720	
		L2 DEC MAC Out of Band Sync Feature with RPR Plus Sup720	
		L2 DEC MAC Out of Band Sync Feature with SSO Sup720	
		L2 DEC Mac Notification Race Sup22	
		L2 DEC Mac Notification Race Sup720	
		L2 DEC No Crash After STP Mode Change Sup720	
		L2 DEC Shut on DEC Port Unicast Flood Sup22	
		L2 DEC Shut on DEC Port Unicast Flood Sup720	
		L2 DEC Switching DEC to DEC Sup720	
		L2 DEC Swithcing DEC to DEC Sup22	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Layer2	Etherchannel, page 3-107	10-Gigabit Ethernet Module Failover	
		10-Gigabit Ethernet Modules L3 Etherchannel Load Balancing	
		Basic L2 Etherchannel Configuration Sup 720	
		Basic L2 Etherchannel Configuration Sup22	
		Basic L2 LACP and Negative IOS to CatOS Sup 22	
		Basic L2 LACP and Negative IOS to CatOS Sup720	
		Basic L2 PAgP and Negative IOS to CatOS Sup22	
		Basic L2 PAgP and Negative IOS to CatOS Sup720	
		Basic L3 Etherchannel Configuration Sup 22	
		Basic L3 Etherchannel Configuration Sup720	
		GE Module Failover Sup22	
		GE Module Failover Sup720	
		L2 Etherchannel Failure and Recovery Sup22	
		L2 Etherchannel Failure and Recovery Sup720	
		L2 Etherchannel Load Balancing	
		L3 10-Gigabit Ethernet Etherchannel Failure and Recovery	
		L3 Etherchannel Failure and Recovery Sup22	
		L3 Etherchannel Failure and Recovery Sup720	
		L3 Etherchannel Load Balancing	
Layer2	Jumbo Frames, page 3-169	Jumbo Frame Support for Unicast Traffic	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Layer2	SPAN, page 3-172	Handling of PIM PDUs Sup22	
		Handling of PIM PDUs Sup720	
		Multicast Receive Only Sup22	
		Multicast Receive Only Sup720	
		Multicast Transmit Only Sup22	
		Multicast Transmit Only Sup720	
		Multicast Transmit and Receive Sup22	CSCsd03035
		Multicast Transmit and Receive Sup720	
		Unicast Receive Only Sup22	
		Unicast Receive Only Sup720	
		Unicast Transmit Only Sup22	
		Unicast Transmit Only Sup720	
		Unicast Transmit and Receive Sup22	
		Unicast Transmit and Receive Sup720	
		rSpan Receive Only Multicast Sup22	
		rSpan Receive Only Multicast Sup720	
		rSpan Receive Only Unicast Sup22	
		rSpan Receive Only Unicast Sup720	
		rSpan Receive and Transmit Multicast Sup22	
		rSpan Receive and Transmit Multicast Sup720	
		rSpan Receive and Transmit Unicast Sup22	
		rSpan Receive and Transmit Unicast Sup720	
		rSpan Transmit Only Multicast Sup22	
		rSpan Transmit Only Multicast Sup720	
		rSpan Transmit Only Unicast Sup22	
		rSpan Transmit Only Unicast Sup720	
Layer2	Trunking, page 3-225	Configuration and Failure Recovery Sup22	
		Configuration and Failure Recovery Sup720	
Layer2	UDLD, page 3-230	UDLD Layer 2 Crossed Fibers	
		UDLD Layer 3 10-Gig Crossed Fibers	
Layer2	VTP, page 3-235	Basic VTP Functionality Sup22	
		Basic VTP Functionality Sup720	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Layer3	BGP, page 3-240	BGP Authentication Failure	
		BGP Basic Functionality	CSCdk65707
		Peer Flapping	
		Redistribution EIGRP to BGP	
		Redistribution OSPF and EIGRP to BGP	
		Redistribution OSPF to BGP	
		Route Flapping with Dampening Configured	
		Route Flapping without Dampening	
		Route Map on Inbound BGP Updates Sup 720	
		Route Map on Outbound BGP Updates with Peer Groups Sup 720	
		Route Scaling	
Layer3	CEF, page 3-265	CEF Packet Switching Sup 22	
		CEF Packet Switching Sup 720	
		CEF uRPF Sup22	
		DFC FIB Consistency Verification Sup 22	
		DFC FIB Consistency Verification Sup 720	
		Forced Software Routing via IP Options Field	
		Traffic Distribution Many Sources to Many Destinations	
		Traffic Distribution Many Sources to One Destination	
		Unicast RPF Sup720	
Layer3	DHCP, page 3-288	DHCP Basic Functionality Sup 22	
		DHCP Basic Functionality Sup 720	
Layer3	EIGRP, page 3-293	Basic EIGRP Functionality	
		EIGRP Authentication Failure	
		EIGRP Neighbor Scaling	
		Redistribution OSPF to EIGRP	
		Route Summarization	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Layer3	IP Multicast, page 3-306	10-GigabitEthernet Module Failover on Auto RP Sup720	
		Auto RP Failover First Hop Router Impact Sup720	
		Auto RP Failover First Hop Router Impact With Reload Sup22	
		Auto RP Failover First Hop Router Impact With Sup Failover Sup22	
		Auto RP Failover RP Impact Sup720	
		Auto RP Failover Traffic Impact Sup720	CSCsl52396
		Basic Auto RP Functionality Sup720	
		Basic IGMP Functionality Sup22	
		Basic IGMP Functionality Sup720	
		Basic Static PIM RP Functionality Sup22	
		Basic Static PIM RP Functionality Sup720	
		Bidir PIM DF Election Sup720	
		Bidir PIM RP Failover RP Impact Sup720	
		Bidir PIM RP Failover Traffic Impact Sup720	
		First Hop Router Functionality Sup22	
		First Hop Router Functionality Sup720	
		GigabitEthernet Module Failover First and Last Hop Router Sup22	
		GigabitEthernet Module Failover First-and Last-Hop Router Sup720	
		GigabitEthernet Module Failover on Auto RP Sup720	
		GigabitEthernet Module Failover with L3 Interface Sup 20	
		GigabitEthernet Module Failover with L3 Interface Sup22	
		IGMP Join and Leave Functionality Sup22	
		IGMP Join and Leave Functionality Sup720	
		IGMP Stress Sup22	
		IGMP Stress Sup720	CSCsm76172
		Last Hop Router Functionality Sup22	CSCsm76172
		Last Hop Router Functionality Sup720	
		Layer 2 GEC Failover Between 102 and Dista-1	
		Layer 2 GEC Failover Between 106 and Dista-2	CSCee21488
		Layer 2 GEC Failover Between 108 and Dista-1	
		Layer 2 GEC Failover Between 110 and Dista-2	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Layer3	IP Multicast, page 3-306	Layer 3 GEC Failover Between 100 and 101	
		Layer 3 GEC Failover Between 100 and 102	
		Layer 3 GEC Failover Between 100 and 104	
		Layer 3 GEC Failover Between 100 and 105	
		Layer 3 GEC Failover Between 100 and 106	
		Layer 3 GEC Failover Between 100 and 97	
		Layer 3 GEC Failover Between 104 and 107	
		Layer 3 GEC Failover Between 104 and 108	
		Layer 3 GEC Failover Between 104 and 109	
		Layer 3 GEC Failover Between 104 and 110	
		MCAST AutoRP Failover FHR Impact with Reload Sup720	CSCee21488
		MSDP SA Delay Sup22	
		MSDP SA Delay Sup720	
		Multicast Stub and Non RPF Rate Limiting Sup22	
		PIM RPF Change Verification Sup720	
		PIM-DR Failover Sup22	
		PIM-DR Failover Sup720	
		Start Receivers then Start Sources	
		Start Sources then Start Receivers	
		Static PIM RP Failover First Hop Router Impact Sup22	
		Static PIM RP Failover First Hop Router Impact Sup720	
		Static PIM RP Failover RP Impact Sup22	
		Static PIM RP Failover RP Impact Sup720	
		Static RP Failover Traffic Impact Sup22	
		Static RP Failover Traffic Impact Sup720	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Layer3	NAT, page 3-473	Addition and Removal of NAT Statements Sup22	
		Addition and Removal of NAT Statements Sup720	
		Dynamic Inside Static Outside Sup22	
		Dynamic Inside Static Outside Sup720	
		Increment Inside Random Outside with Match Host Sup22	
		Increment Inside Random Outside with Match Host Sup720	
		Scaling to 20 000 Translations Sup22	
		Scaling to 20 000 Translations Sup720	
		Static Inside Dynamic Outside Sup22	
		Static Inside Dynamic Outside Sup720	
		Static with 2 Hosts Sup22	
		Static with 2 Hosts Sup720	
		Static with 2 Hosts and Jumbo Frames Sup22	
		Static with 2 Hosts and Jumbo Frames Sup720	
		Static with 2 Hosts and Multicast Sup22	
		Static with 2 Hosts and Multicast Sup720	
		Static with 2 Hosts and UDP Jumbo Frames Sup22	
		Static with 2 Hosts and UDP Jumbo Frames Sup720	
		Static with 2 Networks Sup22	
		Static with 2 Networks Sup720	
		Static with 40 Hosts Extended Sup22	
		Static with 40 Hosts Extended Sup720	
		Static with 40 Hosts Overnight Sup22	
		Static with 40 Hosts Overnight Sup720	
		Static with 40 Hosts Sup22	
		Static with 40 Hosts Sup720	
Layer3	OSPF, page 3-527	Autocost	
		Basic OSPF Functionality	
		Database Verification	
		OSPF Authentication Failure	
		Passive Interface	
		Redistribution EIGRP to OSPF	
		Route Filtering	
Layer3	UDP Broadcast, page 3-548	UDP Broadcast Flooding Sup22	
		UDP Broadcast Flooding Sup720	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Layer3	WCCP, page 3-553	Version 2 Basic Functionality	
Memory Testing	Leaks, page 3-556	Extended Memory Leak Remove and Restore Config Sup22 Remove and Restore Config Sup720 Repeated SSHv1 Sup22 Repeated SSHv1 Sup720 Repeated SSHv2 Sup22 Repeated SSHv2 Sup720 Repeated Telnet Sup22 Repeated Telnet Sup720	
Miscellaneous	Linecard in Supervisor Slot, page 3-575	Linecard in Second Sup Slot Reset Sup720 Linecard with DFC Reset After Write Memory Sup720	
Miscellaneous	System Restart, page 3-580	Power Cycle Software Crash with FTP Core File	
Network Management	IP-SLA, page 3-584	IP SLA IPM Sup720	
Network Management	NDE, page 3-596	NDE Basic Functionality Sup22 NDE Basic Functionality Sup720	
Network Management	NTP, page 3-601	Basic NTP Functionality Sup22 Basic NTP Functionality Sup720 Server Failover Sup22 Server Failover Sup720	
Network Management	SNMP, page 3-610	Basic SNMP Functionality Sup22 Basic SNMP Functionality Sup720 Copying via SNMP to Running Config Sup22 Copying via SNMP to Running Config Sup720 SNMP Config Sync Sup22 SNMP Config Sync Sup720 SNMP Malformed Packet SNMP Walk of DUT Sup22 SNMP Walk of DUT Sup720	
Network Management	Syslog, page 3-629	Syslog Basic Functionality Sup22 Syslog Basic Functionality Sup720	
Network Management	TACACS, page 3-633	User Authentication Sup22 User Authentication Sup720	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Network Operation	Baseline, page 3-638	Baseline Network Steady State	
		Baseline Network Validation	
Network Operation	Resiliency, page 3-643	Rapid Link Flap	
		Verification of Network Function and Resiliency	
QoS	IDC, page 3-648	Inband Ping Functionality Sup22	
		Inband Ping Functionality Sup720	
		Overnight Stress Sup22	
		Overnight Stress Sup720	
		QoS Basic Functionality Sup 720	
		QoS Basic Functionality Sup22	
		QoS Effects on HSRP Functionality Sup22	
		QoS Effects on HSRP Functionality Sup720	
		QoS Effects on OSPF Functionality Sup22	
		QoS Effects on OSPF Functionality Sup720	
		QoS Effects on STP Functionality Sup 22	
		QoS Effects on STP Functionality Sup720	
Redundancy	Power Supply, page 3-677	Power Supply Failure 2500W Sup22	
		Power Supply Failure 2500W Sup720	
		Power Supply Failure 6000W Sup22	
		Power Supply Failure 6000W Sup720	CSCsc81109

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Redundancy	Supervisor, page 3-687	Change Switching Modes Basic Function Sup22	
		Change Switching Modes Basic Function Sup720	
		Change Switching Modes with RPR Plus Failover Sup22	
		Change Switching Modes with RPR Plus Failover Sup720	
		Change Switching Modes with SSO Failover Sup22	
		Change Switching Modes with SSO Failover Sup720	
		Compact Mode RPR Plus Failover Sup22	
		Compact Mode RPR Plus Failover Sup720	
		Compact Mode SSO Failover Sup22	
		Compact Mode SSO Failover Sup720	
		Compact Mode Standby Sup Reset Sup22	
		Compact Mode Standby Sup Reset Sup720	
		Failover RPR+ Sup22	
		Failover RPR+ Sup720	
		Failover SSO NSF Sup 720	
		Failover SSO NSF Sup22	
		Reliability Standby Sup Repeated Reset Sup720	
		Sup Hot Insert Sup22	CSCsj67108
		Sup Hot Insert Sup720	
		Truncated Mode RPR Plus Failover Sup22	
		Truncated Mode RPR Plus Failover Sup720	
		Truncated Mode SSO Failover Sup22	
		Truncated Mode SSO Failover Sup720	
		Truncated Mode Standby Sup Reset Sup22	
		Truncated Mode Standby Sup Reset Sup720	
Security	802.1x, page 3-746	802.1x Authentication with EAP and MD5 Sup2	
		802.1x Authentication with EAP and MD5 Sup720	
		802.1x with EAP and MD5 Negative Sup22	
		802.1x with EAP and MD5 Negative Sup720	
		Bad TACACS Login Sup22	
		Bad TACACS Login Sup720	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Security	ACL, page 3-761	ACL 100 Sup22	
		ACL 100 Sup720	
		ACL 101 Sup22	
		ACL 101 Sup720	
		ACL 110 Sup22	
		ACL 110 Sup720	
		ACL 131 Sup22	
		ACL 131 Sup720	
		Program TCAM with Large ACLs Sup720	
Security	Port Scan, page 3-780	NMAP Port Scan Sup22	
		NMAP Port Scan Sup720	
Security	SSH, page 3-784	SCP File Transfer Sup720	
		SSH Client Basic Functionality Sup720	
		SSH Client Five Sessions Sup720	
		SSH Client Max VTY Sup720	
		SSH Vulnerability Sup22	
		SSH Vulnerability Sup720	
		Telnet to SSH Client Max VTY Sup720	CSCsm41952
Upgrade	Software, page 3-801	Fast Software Upgrade Current to Latest IOS Release Sup22	
		Fast Software Upgrade Current to Latest IOS Release Sup720	
		Fast Software Upgrade SXE5 to Latest IOS Release Sup720	
		Fast System Upgrade SXB11 to Latest IOS Release Sup22	
		Fast System Upgrade SXB11 to Latest IOS Release Sup720	
		Upgrade Current to Latest IOS Release Sup22	
		Upgrade Current to Latest IOS Release Sup720	
		Upgrade SXB11 to Latest IOS Release Sup22	
		Upgrade SXB11 to Latest IOS Release Sup720	
		Upgrade SXE5 to Latest IOS Release Sup720	

Table K-7 Cisco Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
User Interface	Command Line, page 3-818	Parser RP Platform Commands via SSH Sup22	
		Parser RP Platform Commands via SSH Sup720	
		Parser RP Platform Commands via Telnet Sup22	
		Parser RP Platform Commands via Telnet Sup720	
		Parser RP via SSH BGP Commands Sup720	
		Parser RP via SSH Commands A to L Sup22	CSCsc81109
		Parser RP via SSH Commands A to L Sup720	CSCsc81109
		Parser RP via SSH Commands M to Z Sup22	
		Parser RP via SSH Commands M to Z Sup720	
		Parser RP via SSH EIGRP Commands Sup22	
		Parser RP via SSH EIGRP Commands Sup720	
		Parser RP via SSH OSPF Commands Sup22	
		Parser RP via SSH OSPF Commands Sup720	
		Parser RP via SSH RIP Commands Sup22	
		Parser RP via SSH RIP Commands Sup720	
		Parser RP via Telnet BGP Commands Sup720	
		Parser RP via Telnet Commands A to L Sup22	
		Parser RP via Telnet Commands A to L Sup720	
		Parser RP via Telnet Commands M to Z Sup22	
		Parser RP via Telnet Commands M to Z Sup720	
		Parser RP via Telnet EIGRP Commands Sup22	
		Parser RP via Telnet EIGRP Commands Sup720	
		Parser RP via Telnet OSPF Commands Sup22	
		Parser RP via Telnet OSPF Commands Sup720	
		Parser RP via Telnet RIP Commands Sup22	
		Parser RP via Telnet RIP Commands Sup720	

Secure Socket Layer Module (SSLM) 3.1.1

The following Safe Harbor Secure Socket Layer Module (SSLM) Release 3.1.1/12.2(18)SXF was tested and certified as a conditional recommendation.

Table K-8 summarizes testing executed as part of the Cisco Secure Socket Layer Module (SSLM) Release 3.1.1/12.2(18)SXF Safe Harbor initiative. This table include component tests for each feature or function.

Table K-8 Cisco Safe Harbor Test Results Summary

Features/Functions	Tests	Results
Secure Socket Layer (SSL)	Upgrade	CSCek28184, CSCek28201, CSCek28145, CSCek27840, CSCek28344
	CLI Parser	
	Manual Certificate Signing Request	CSCek27840, CSCeh81581
	Certificate and Key Importation with PEM Paste	
	Graceful Certificate Rollover	CSCek27840
	URL Rewrite	CSCec74017, CSCek29512, CSCek29525
	SSL Client Proxy Services	CSCek28849
	Client Authentication	CSCek28849, CSCek30466
	HTTP Header Insert Policy Client IP Port	
	HTTP Header Insert Policy Client Cert	
	HTTP Header Insert Custom Header	CSCek29512
	HTTP Header Insert Session	
	HTTP Header Insert Policy ALL	CSCek29512
	SSL Termination	
	SSLSM Configuration Virtualization	New
	Export Ciphers	New
	Protected Key Storage	New, CSCee54025, CSCek31540, CSCek31534
	SSL Session Auto Renegotiation	New
	Maximum Connections	New, CSCek31276, CSCek31281, CSCek31283

Wide Area Application Services (WAAS) Release 4.0.13.23

The following Safe Harbor Wide Area Application Services (WAAS) Release 4.0.13.23 was tested and certified as a conditional recommendation.

Table K-9 summarizes testing executed as part of the Cisco Wide Area Application Services (WAAS) Release 4.0.13.23/12.2(18)SXF9 Safe Harbor initiative. This table include component tests for each feature or function.

Table K-9 Cisco Safe Harbor Test Results Summary

Test Suites	Feature/Function	Tests	Results
Basic Functionality	Authentication, page 8	Radius Authentication	
	Base Configuration, page 10	ACL Configuration	CSCsk51367
		CM Configuration	
		NTP Configuration	
		Restore Factory Defaults	
		Core WAE Configuration	
	Upgrade, page 18	WAE Software Upgrade	
Policy Based Routing	Acceleration, page 20	PBR HTTP Acceleration	
	Configuration, page 23	PBR Configuration	
SNMP Monitoring	SNMP Polling, page 24	SNMP Poll	
WAFS	CIFS Performance, page 25	WAFS Cache Hit T1 80ms	
		WAFS Cache Miss T1 80ms	
		WAFS Native WAN Access T1 80ms	
	WAFS Configuration, page 30	WAFS Baseline Configuration T1 80ms	
	Preposition, page 32	WAFS FAT32 Preposition T1 80ms	CSCsj99861
		WAFS NTFS Preposition T1 80ms	
WCCP L2 Redirection	WCCP Acceleration, page 36	FTP Acceleration L2 Redirect	
		HTTP Acceleration L2 Redirect	
		SMTP Acceleration L2 Redirect	
	WCCP Configuration, page 39	L2 Redirection Configuration	
	Load Testing, page 40	Maximum Throughput	
		Overload Bypass	
	WCCP FARM, page 42	Modify Mask Assignment	
		Modify WCCP Through CLI	CSCsk01950
		Multiple WAE Failure	
		Single WAE Failure	
		Single WAE Graceful Shutdown	CSCsk01950

APPENDIX **L**

DCAP 4.0 DDTS Bugs

The DCAP test engineering team logs all

- Bug defects initially identified and subsequently created, or filed, during the testing cycle
- Existing bugs encountered during the testing cycle
- Existing bugs of interest but not encountered during the testing cycle, and
- Existing bugs previously encountered and not fixed during the testing cycle

Development Defect Tracking System (DDTS) software bugs, with descriptions and severity are available for consideration, listed by the following technologies, and sorted by severity then number.

Volume 2: LAN (Layer 2-3) Infrastructure

The Cisco DCAP 4.0 LAN infrastructure is built around the Catalyst 6500 switching platform that provides for various features such as 10-Gigabit Ethernet connectivity, hardware switching, and distributed forwarding. While testing focuses on the Catalyst 6500 platform, the Catalyst 4948-10GE switch is also deployed to provide top-of-rack access to data center servers. The LAN infrastructure design is tested for both functionality and response to negative events.

The following DDTS types were logged:

- L2-3 CSM DDTS Encountered, page L-2
- L2-3 CSM DDTS of Interest but Not Encountered, page L-2
- L2-3 ACE DDTS of Interest but Not Encountered, page L-5

L2-3 CSM DDTS Encountered

Table L-1 lists L2-3 CSM DDTS Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-1 DDTS Encountered in DCAP 4.0 L2-3 CSM Testing

DDTS	Description	Severity
CSCsk60108	Cat 6k has traceback during reload;	3
CSCsc81109	show crypto pki server cli command causes Traceback;	4

L2-3 CSM DDTS of Interest but Not Encountered

Table L-2 lists L2-3 CSM DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-2 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 CSM Testing

DDTS	Description	Severity
CSCsg07870	crash seen on switchover at pf_redun_sync_port_asic_on_swover.	1
CSCsi99869	Bus error crash following %MCAST-SP-6-GC_LIMIT_EXCEEDED.	1
CSCsj27811	EOBC buffer leak caused by CMM module.	1
CSCsl04908	WCCP: shutdown of appliance i/f leads to c6k reload	1
CSCei52830	Banner command sync is broken by CSCin86483.	2
CSCsi00706	Sierra: upon fib tcam exception to use ratelimiter and not reload	2
CSCsi29875	3/27: SP: oir_rf_reload_self: icc_req_imm failed, node not booting	2
CSCsi51649	RP crashes@fm_send_inband_install_message+21C in many cases with NAT	2
CSCsi94863	New xenpak background task.	2
CSCsj56703	SSO failover causes RSTP forwarding and physical interfaces blocking	2
CSCsj68911	DFC mem leak in SP Logger Proces when redundancy force-switchover issued	2
CSCsj83966	Syslog traps cause CPUHOG when lot of interface come up at same time. .	2
CSCsj92874	Catalyst 6500 May Not Send linkup/linkdown SNMP Traps and may reload	2
CSCsk09302	CDP packets not received on WS-6704-10GE/CFC links with MLS QoS enabled	2
CSCsk21414	NAC : Buffer leak in small buffer pool .	2
CSCsk27835	Disable unsupported service modules in SXF Software Modularity images	2
CSCsk33724	DOM does not work anymore for cwdm gbic/sfp	2
CSCsk33740	replay window size of 1024 causes IPSec Policy Check and Replay Failure	2
CSCsk39022	Modular IOS: ip directed-broadcast not working	2
CSCsk41134	ISAKMP SA neg not successful for in tunnel mode w/ RSA-SIG	2
CSCsk41134	ISAKMP SA neg not successful for in tunnel mode w/ RSA-SIG	2
CSCsk55423	7600's SPD implementation allow COS 5 or above in Extended headroom	2
CSCsk57789	SUP720-3BXL CPU Often Spikes to 100% When Upgraded From 18SXE5 to 18SXF9	2
CSCsk60874	show tech needs 'show diagnostic results' and 'show diagnostic events' .	2
CSCsk63794	FlexWAN WS-X6582-2PA + T3+ Serial PA may crash/reload	2
CSCsk80935	SXF12, SNMP response being broadcast .	2
CSCsk80935	SXF12, SNMP response being broadcast .	2
CSCsk80935	SXF12, SNMP response being broadcast .	2
CSCsk82191	Multicast groups sometimes not populated on switchover.	2
CSCsl00130	GRE tunnel not HW accelerated after reboot when source from HSRP address	2
CSCsl07297	SXF11: BGP no neighbor command caused Address Error exception .	2
CSCsl07297	SXF11: BGP no neighbor command caused Address Error exception .	2
CSCsl21106	Tunnel destination command crashes MSFC running in hybrid mode .	2
CSCsl32122	Remote Access for certificate users fails during mode config	2
CSCsl61086	urpf global disable even some intf with urpf	2
CSCsl63311	6500 May Experience High CPU due to NAT traffic	2

Table L-2 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 CSM Testing (continued) (continued)

DDTS	Description	Severity
CSCsl65335	WCCP: reload following ACL update	2
CSCsl70403	Egress policy is not supporting on interface with mpls config	2
CSCsl79714	Crash ip csg with service timestamps log datetime msec localtime show	2
CSCsm06740	Memory Leak in AAA accounting and Virtual Exec	2
CSCsm12247	WCCP: hash assignment may be lost after service group change	2
CSCsm48398	mls cef adj leaking	2
CSCsm57407	c7600-ssc-400 : VPN-SPA reset due to hyperion asic reset (in 7603)	2
CSCsm58533	disable wccp crashs CAT-6zK with Sup-720 operation	2
CSCsm65726	Sup22: Changing from L2 DEC to L3 DEC causes software switching	2
CSCee77416	changing logg-buffer cause GW to print no-mem and tracebacks.	3
CSCsd11258	Sh env display pwer input/output presence even on pwrinput cable removal	3
CSCsd90173	TestIPSecEncrypDecrypPkt HM test config init error reporting is needed	3
CSCsg81380	EIGRP connected interface is not advertised to neighbor after a reload	3
CSCsj00385	logging event link-status default negates existing interface config	3
CSCsj09045	GRE keepalives don't process correctly, tunnel interface flap.	3
CSCsj16292	DATACORRUPTION-1-DATAINCONSISTENCY: copy error .	3
CSCsj45031	Cat6k unable to SCP files from Tectia ssh server	3
CSCsj56102	Upgrade of DFC rommon fails in 12.2SX train IOS	3
CSCsj72529	WS-X6748-GE-TX:Link may not come up on initiating error recovery patch	3
CSCsj74309	Crash on secondary switch in pair when primary switch is reloaded	3
CSCsj77819	After SSO traffic is punted to the CPU for 20 seconds	3
CSCsj89305	RADIUS/NAS-IP address is sent out as 0.0.0.0	3
CSCsk19652	Failed to assert Physical Port Administrative State Down alarm	3
CSCsk33045	MST BPDU *must* be sent untagged, even when the switch is configured wit	3
CSCsk43035	On PA-8T-V35 QoS conformed counter stops when child policy value changed	3
CSCsk53642	RSVP PATH msg not forwarded to MCAST receiver .	3
CSCsk57258	Pkg punt to CPU and bounce back When EoMPLS MTU Failure	3
CSCsk58040	WS-X6148A-GE-45AF retains previous modules MACs after OIR	3
CSCsk59181	Add static NAT will interrupt normal PAT traffic	3
CSCsk60108	Cat 6k has traceback during reload	3
CSCsk61065	DSCP value incorrectly reset for mcast packets on PE router	3
CSCsk84944	unidirectional Ethernet UDE is broken on WS-6704 after SW upgrade	3
CSCsl08952	rapid link changes causes memory leak on sup32 int with service policy	3
CSCsl09867	Exec-timeout not working at more prompt when using Modular IOS	3
CSCsl09867	Exec-timeout not working at more prompt when using Modular IOS	3
CSCsl27957	%EC-SP-5-CANNOT_BUNDLE2 is seen after inserting WS-SVC-FWM-1	3

Table L-2 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 CSM Testing (continued) (continued)

DDTS	Description	Severity
CSCsl34391	Output of 1st page of sh crypto ipsec sa is blank	3
CSCsl41784	ION: ARP Input memory leak with mobile ip arp	3
CSCsl47365	TACACS+ authorization should ignore unknown attribute	3
CSCsl56805	router keep replying to ping destined ff02::1	3
CSCsl57645	tacacs-server directed-request fails for enable authentication on 6500	3
CSCsl78586	bgp-bw load sharing does not work when appended community	3
CSCsl82199	Vacl capture - CMM - Medusa FPOE_CAP2 reg not set properly	3
CSCsl86261	Bridge Group output not being displayed	3
CSCsl98238	QoS statistics-export only exports to directly-connected destinations	3
CSCsm06821	vpnv4 update from a RR can not trigger the vrf import process	3
CSCsm17466	WLSM : dhcp ack not seen by dhcp snooping with ATM transport	3
CSCsm29181	Crash when NBAR applied to sub-interface	3
CSCsm31178	policy-map stops working on a good int if wrongly applied on another int	3
CSCsm32363	Netflow SLB sw-installed entries not aging out	3
CSCsm32363	Netflow SLB sw-installed entries not aging out	3
CSCsm37351	Router is not forwarding (s,g) prune	3
CSCsm38160	Multicast Traffic from source in PVLAN does not get routed properly	3
CSCsm41952	SSH session hangs intermittently	3
CSCsm49062	cwan2: show queueing interface reports double count for wfq drops	3
CSCsm57580	cat6k/mpls - mpls adj is not removed from int when mpls disabled on int.	3
CSCsm59500	ciscoEnvMonSupplyState is normal even when PS is OFF	3
CSCsm62153	Packet Buffer Capture May Crash a 6500 in IOS	3
CSCsm63524	SUP32 crashes due to SP hang when it recovers from errdisable	3
CSCsm63524	SUP32 crashes due to SP hang when it recovers from errdisable	3
CSCsm64998	Catalyst 6500 may experience scp_dnld tracebacks during reload	3

L2-3 ACE DDTS of Interest but Not Encountered

Table L-3 lists L2-3 ACE DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-3 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 ACE Testing

DDTS	Description	Severity
CSCsi44096	feature CONN-REUSE not functioning without failed sessions	1
CSCsj38511	HM:while running scripted probes test case 11.2.1.3, ACE get a core	1
CSCsj38645	SSL init stress using RC4-MD5 crashes ACE	1
CSCsj55836	Buffer leak / crash with FE-SSl traffic with L7 features configured	1

Table L-3 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 ACE Testing (continued) (continued)

DDTS	Description	Severity
CSCsk30865	X_TO_ME: Hung while running FE with client auth with 4k bits key	1
CSCsk35523	cfgmgr crash during config/unconfig of 250 contexts and ft groups	1
CSCsk48871	Kernel Crash on trying on doing a tftp copy soon after bootup	1
CSCsk49756	ACE NP Crash with FE-SSL trafific having Match-src as class	1
CSCsk50946	ssl is crashed when applying a chaingroup with 8 certificates	1
CSCsf09910	ACE Conn Repl: FTP Data channel stalls after swtichover with inspection	2
CSCsh19664	scripted probe uses all resources and then causes all probes to fail	2
CSCsi36216	traffic will not go to one particular rserver	2
CSCsi69018	sticky database not in sync after ft switchover with ACE in bridge mode	2
CSCsi70738	config will not sync running 2 ACEs in same chassis in bridge mode	2
CSCsi79974	Syslog not sending CP or DP level 6 messages.	2
CSCsj12610	xml script doing 'show version' on ACE causes silent reload after 30 hrs	2
CSCsj14813	SSL connections fail after changing the certs and Keys with traffic	2
CSCsj16667	HM crash when rserver put inservice after long period of being OOS	2
CSCsj17220	L4Lb with IP sticky and NAT under stress causes many ME's to be stuck	2
CSCsj24815	Standby ACE crashed during A1(4n) to A1(5) throttle image upgrade	2
CSCsj25440	ACE/Scimitar:no radius/tacacs+/ldap failed	2
CSCsj25941	ACE - ACE reboot w/core NP 0 Failed : NP ME Hung during Fragmenting	2
CSCsj26023	QNX qconn process core	2
CSCsj37029	A15: Telia SSL performance test case does not pass	2
CSCsj37202	buffer leaked after running front-end ssl traffic for 24 hours	2
CSCsj37653	HTTP: Wrong deobfuscation of multibyte UTF8 encoding in URL	2
CSCsj56916	L7 Connectings failing with LB policy misses	2
CSCsj59659	ACE crash with Mutiple ME's stuck with L7 Lb traffic	2
CSCsj74391	ACE does not g-ARP NAT Address upon reload	2
CSCsj77194	Intermittent erroneous XML response for show context in Admin context.	2
CSCsj88500	<show context> & <show service-policy> display incomplete/truncated info	2
CSCsj95609	TCP: TCB and other TCB has same reassembleq	2
CSCsk03039	malformed xml returned from show_context	2
CSCsk07539	Stress:Multiple ME stuck with SSL-INIT traffic havig multiple L7 Feature	2
CSCsk26606	ACE: malformed SSL record if certificate chain length higher than 4k	2
CSCsk30079	HA flapping while running FE SSL with client auth perf test	2
CSCsk30714	internal buffers leaked after running FE ssl with client-auth	2
CSCsk33251	ACE: IP Source stickiness can fail after upgrade and/or failover	2
CSCsk35058	Initial TCP sessions are dropped after a reboot	2
CSCsk35629	unable to login. Internal CLI error: No such file or directory	2

Table L-3 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 ACE Testing (continued) (continued)

DDTS	Description	Severity
CSCsk36611	ACE: ssl re-handshake fails with IE when cert chain bigger than 4k byte	2
CSCsk44718	ace core dump after sup sso failover	2
CSCsk50407	ACE breaks application by performing implicit PAT on loadbalanced conns	2
CSCse70005	tcp port numbers not converted correctly with csm2ace tool	3
CSCse70052	csm2ace tool: maxconns not converted as conn-limit in serverfarm	3
CSCse70823	csm2ace tool: does not convert serverfarm predictor	3
CSCse70865	csm2ace tool: serverfarm real probe not applied to ACE config	3
CSCse72427	csm2ace tool: places serverfarm probe in wrong location	3
CSCse90603	auto complete doesn't work on checkpoint names	3
CSCsg06046	persistent-rebal causes all persistent gets to be remapped on the server	3
CSCsg06046	persistent-rebal causes all persistent gets to be remapped on the server	3
CSCsg10024	ftp L7 denies cdup but not cd ..	3
CSCsg23016	class map insert-before doesn't work on previously configured class map	3
CSCsg23096	show conn serverfarm not showing ftp data connection	3
CSCsg52355	RHI Injected routes lost after SUP switchover	3
CSCsg70663	csm2ace tool - csm config should be ported to ACE usr context, not Admin	3
CSCsg70682	csm2ace tool - nat not supported	3
CSCsg70686	csm2ace tool - naming convention for resource class + undefined objects	3
CSCsg70689	csm2ace tool - no alert to the user of a script not being present on ace	3
CSCsg70696	csm2ace tool - converstion of multi-policy to ace is not correct	3
CSCsg70702	csm2ace tool - class-map inconsistency	3
CSCsg70709	csm2ace tool - loadbalance vip icmp-reply active (needs to be added)	3
CSCsg70719	csm2ace tool - inconsistent naming convention (case sensitive)	3
CSCsg70731	csm2ace tool - sticky config on csm only partically converts to ace	3
CSCsg70736	csm2ace tool - command <replicate sticky> NOT <replicate-sticky> on ace	3
CSCsg70742	csm2ace tool - description feild (vlan) not converted fom csm to ace	3
CSCsg71636	csm2ace tool - incorrect cli command for ace class-map (dns vs domain)	3
CSCsg71838	csm2ace tool - invalid regular expression (conversion from csm to ace)	3
CSCsg73137	csm2ace tool - csm port on probe is inherited from the vservers not ACE	3
CSCsg73181	cms2ace tool - ACE conversion via CSM omits last for regex string	3
CSCsg73521	csm2ace tool - static arp entry on csm not listed in unsuported commands	3
CSCsg73557	csm2ace tool - capp udp optoins unsupported comand on csm omitted	3
CSCsg78690	sh tech returns errs for all CLIs that the user's role denies access to	3
CSCsg80474	SSL show stats commands missing - only ssl stats are ref in ucdump	3
CSCsg80625	spanned udp request (not frag) packets are dropped by ace	3
CSCsg80625	spanned udp request (not frag) packets are dropped by ace	3

Table L-3 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 ACE Testing (continued) (continued)

DDTS	Description	Severity
CSCsg86626	after upgrade failure unable to boot prior image from rommon	3
CSCsg88014	copy ftp image fails when overwriting the same filename	3
CSCsg91127	license mismatch after member is added to context	3
CSCsg98014	1(Error: resources in use, attempting to add a RC to a context	3
CSCsh06784	Ingress vlan SPAN send all vlan to destination port	3
CSCsh20546	maxconn, persistence and sticky not working together	3
CSCsh27746	tacacs not working with Cisco Secure ACS	3
CSCsh53696	resource usage show current value greater than max allowed	3
CSCsh53739	rserver shows arp_failed, but probe, arp and access exists	3
CSCsh72916	poor performance with persistent rebalance and url hash load balancing	3
CSCsh81195	total conn-failures incrementing when client sends reset on estab conn	3
CSCsh82790	client request delay if server mtu is set to 250	3
CSCsh91414	rhi routes not injected or removed quickly after forced ft failover	3
CSCsi13650	persistent HTTP 1.1 conn that reaches max-conn limit gets reset	3
CSCsi20428	current conn counter on serverfarm shows active conn with no flows	3
CSCsi23858	probe failure when changing config	3
CSCsi26014	counters not incrementting when exceeding parse length	3
CSCsi43733	scripted tftp probe shows failures	3
CSCsi43733	scripted tftp probe shows failures	3
CSCsi52002	sticky warning log message after reload	3
CSCsi60741	Error 12459: :[12459] Login failed	3
CSCsi62289	ace does not decrement the ttl when packets route through it	3
CSCsi80046	tac-pac cmd issues	3
CSCsi81930	PSS error while configuring snmp-server community in client context	3
CSCsi82962	Certain incoming packet data causes ACE to lose MSS setting mid-flow.	3
CSCsi88102	mts error mts_acquire_q_space() failing - no space in sap 2914	3
CSCsi92341	ACE sh logging may not show latest log	3
CSCsi93631	ACE capture feature may disregard bufsize limit.	3
CSCsi98829	SSL handshake fails using self signed certificate	3
CSCsj04439	Configuring no logging message 313004 shows up in config twice.	3
CSCsj04447	domain checks for scripted probes are missing	3
CSCsj04464	domain checking for monitoring sticky database not working	3
CSCsj07054	capture command with circular-buffer option not overwriting the buffer	3
CSCsj08692	Stanby Blade crashed on reloading Active with L3/L7/SSL traffic	3
CSCsj13557	Sticky database entries of active connections disappear after HA upgrade	3
CSCsj15501	'aaa' config in Admin context shows up in user context	3

Table L-3 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 ACE Testing (continued) (continued)

DDTS	Description	Severity
CSCsj20156	ACE tac-pac generation may fail with `No space left on device` message.	3
CSCsj20171	Cannot remove 'service policy input global'	3
CSCsj21032	Improve performance of crb_compute_diff()	3
CSCsj21178	itasca_ssl(911) has terminated on receiving signal 11, key import	3
CSCsj23996	acl download failure due to duplicate lineno	3
CSCsj24580	user ft group stuck in bulk_sync because of missing arp	3
CSCsj25054	CMCLOSE and TCP stuck with Pipelined traffic and rebalance	3
CSCsj26329	SNMP walk shows low out packet for Baikal compared to the device itself	3
CSCsj27746	golden: ftp data transfer hang after ft switchover	3
CSCsj27882	Golden: ha_basic fails in test 5.10. FT group in STANDBY_COLD	3
CSCsj28335	Poor throughput with L7LB and persistence rebalance	3
CSCsj30048	ACE:ssh key is not synced to STANDBY	3
CSCsj30713	ping to some of the vips fails after policy-map config change	3
CSCsj31419	ACE reset by SUP due to keepalive polling failure	3
CSCsj34419	Replicate connection if xlate error counter is increase	3
CSCsj35046	checkpoint rollback issues with banner and hostname/ssh keys	3
CSCsj37212	crashinfo created	3
CSCsj40904	tcp unproxy can fail with unproxy_ack w/fin flag set	3
CSCsj41909	ACE: NP 1 Failed : NP Process: QNX process io_net Crashed	3
CSCsj43225	ACE: Sticky feature issues after downgrading from A1(5a) to A1(4n)	3
CSCsj47861	Demo license install failure	3
CSCsj48884	ACE: ftp data channel fixup must have priority over policy nat	3
CSCsj49249	server info isn't updated while learning the same cookie from other srv	3
CSCsj51148	Elapsed time is not taken into account while changing the sticky timer	3
CSCsj51161	NAT policy in DP not updated if nat pool removed	3
CSCsj51637	xml agent should do roll back when error occur	3
CSCsj52843	Syslog for rserver state change due to the probe failure not generated	3
CSCsj53114	rserver's sticky is removed while rejecting the conns. due to MAXCONNS	3
CSCsj55058	Existing connection is not closed while changing the sticky serverfarm	3
CSCsj58598	ME core with ICM util at 100%	3
CSCsj61121	ACE is not learning the multiple instances of the same cookie	3
CSCsj62850	Disabling timeout activeconns still flushes active conns. sticky entry	3
CSCsj63564	HA: auto-sync not cleared when context ID is reused.	3
CSCsj64177	sfarm total conn counter is updated for each cookie request in same conn	3
CSCsj64723	ACE: http/https probe no hash does not clear the reference hash value	3
CSCsj64822	ace needs process to track and provide supplied probe script updates	3

Table L-3 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 ACE Testing (continued) (continued)

DDTS	Description	Severity
CSCsj64833	UDP/TCP traceroute does not work to configured ACE IP Interfaces	3
CSCsj65569	ACE: LB Policy miss with match souce-address clause in L7LB policy	3
CSCsj65699	No Syslogs are getting generated while Disabling probe manually	3
CSCsj66157	TCP Reset from client not processed	3
CSCsj68643	can_wait_specific_msg: Aborting call (SAP 27, pid 959)	3
CSCsj70522	log messages not being displayed on console or in buffer	3
CSCsj70624	TCP reset during HTTP traffic	3
CSCsj74004	Issue with adding nat dynamic and nat-pool to an empty service policy	3
CSCsj74250	Key not encrypted when Configuring TACACS on ACE	3
CSCsj74292	ACE Subcontext Configuration Lost with FT Config & No Peer	3
CSCsj74518	NAT dnld err when removing a nat-pool that has multiple IP ranges	3
CSCsj80265	SSHv1 with TACACS sessions into ace fail	3
CSCsj84841	HTTP: Improve reproxy performance of chunked response	3
CSCsj85316	ICMP Err Inspection: Translate Real IP to VIP in IP header of ICMP pkt	3
CSCsj89960	ACE - generates error when using global NAT address with diff ports	3
CSCsj90095	ooo out-of-order packets in response sent from ace to client	3
CSCsj91529	Incorrect syslog is generated while deleting the class map.	3
CSCsj91836	Trinity: RTSP Inspect adjusting content-length of SDP msg incorrectly	3
CSCsj93358	VIP has to be in inservice even though all rservers are in MAXCONNS stat	3
CSCsj94366	unable to change console settings	3
CSCsj95752	MAXCONNS rserver's total conn counter is updated for each request	3
CSCsj97786	Removing service-policy from one vlan deletes from all	3
CSCsj99175	Deletion failed for Sfarm Probe Table	3
CSCsj99704	Regex resource clas minimums not guaranteed in contexts	3
CSCsk00341	Unconfigured static cookie shown in sh run & still provides stickness	3
CSCsk01206	SSH key generation message and no keys generated	3
CSCsk02229	Ephemeral step up for SSL initiation does not work	3
CSCsk03514	ACE crashes during normal operation	3
CSCsk08296	Changing L7 policy disrupts existing traffic flows	3
CSCsk10045	current conn. counter is not all decremented with server-conn reuse	3
CSCsk10290	Different interface Id's for Active/Standby cause xlate error.	3
CSCsk15979	idle connection closed before timeout	3
CSCsk16083	https probe using hash always show success with Windows 2000	3
CSCsk16823	syslog buffer problems when increasing resource limit	3
CSCsk19394	static sticky entries aren't generated for rservers in backup serverfarm	3
CSCsk21143	Connections lost with ft preempt and heartbeat lower than 200ms	3

Table L-3 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 ACE Testing (continued) (continued)

DDTS	Description	Severity
CSCsk23763	Global tacacs key configuration doesn't work	3
CSCsk25291	conns. are not load-balanced to the rserver returned from MAXCONNS state	3
CSCsk25567	Uninstall contexts license should check the numbers of contexts	3
CSCsk26767	ACE: can't delete or modify chaingroup even if not being used	3
CSCsk28161	sh tech o/p partially xmlized	3
CSCsk31782	high CPU utilization running large number of probes	3
CSCsk32367	backup rserver's weight should not be included for weight normalization	3
CSCsk33385	Spontaneous Crash with Fastpath Stuck	3
CSCsk33613	can't rollback from load-balance policy-map to http load-balance policy	3
CSCsk34767	unable to clear ftp inspect stats	3
CSCsk34831	ACE HealthMon (HM) corefile cannot be loaded in gdb	3
CSCsk34843	Data Bus error messages on the console	3
CSCsk34907	incremental sync failure but config is in sync	3
CSCsk34973	ftp inspect policies remain after removing class-map from policy-map	3
CSCsk36590	ACE: QNX core in LbUtil_ServerGoodListCheck	3
CSCsk37369	Error in Message receive : Interrupted function call	3
CSCsk38667	ACE fails to send window probe to host sending window 0 after conn. estb	3
CSCsk39152	ACE: cat: write error: No space left on device	3
CSCsk42541	Error: Called API encountered error	3
CSCsk42839	sticky resource not discovered for this context -- type 85	3
CSCsk43066	HA state STUCK in TL_SETUP, after upgrading to A161	3
CSCsk43321	system crash while generating an IXP crash corefile.	3
CSCsk46504	ACE experienced an ICM microengine core during normal operation A1(4L)	3
CSCsk46544	logging monitor is not synched to standby	3
CSCsk46771	keys with file name more then 64 characetrs messed up the sh crypto comm	3
CSCsk49388	csr with 2048 bit key is generated with some extra characters	3
CSCsk49666	RTSP Connection hangs after switchover with inspection and Nat	3
CSCsk51091	HTTP Error Code Monitoring breaks https to WAAS Farm	3
CSCsk51599	check-point rollback is not clearing interface vlan config	3
CSCse90570	unable to break out of copy command with ctrl-c	4
CSCsg07468	service-policy packet counter wrong when using tcp-reuse	4
CSCsg07544	service-policy hit counter is misleading	4
CSCsg07570	show service-policy does not show advanced-options that are configured	4
CSCsg23578	when clearing stats the command stats should be used	4
CSCsg24832	quickly adding and removing ftp inspect policy cause policy error messag	4
CSCsg42659	1 connection counts as 2 on various show commands	4

Table L-3 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 ACE Testing (continued) (continued)

DDTS	Description	Severity
CSCsg58753	show sticky database has a difficult to read format	4
CSCsg58769	show sticky database lacks important detail	4
CSCsg58783	show sticky database or with group does not display static entries	4
CSCsg75273	show conn serverfarm does not have detail argument	4
CSCsg75308	show serverfarm <name> detail should include some probe information	4
CSCsg88045	copying file to image transfer mode is not enforced	4
CSCsg92839	ssl probe - unable to select 5 ciphers	4
CSCsg93314	unable to delete an rsa key when name is >= to 86 char.	4
CSCsh12734	ACE-1-727005: HA: Configuration Replication failed needs more detail	4
CSCsh12819	no log message when ft auto-sync executed on standby	4
CSCsh13030	unable to paste a small config through telnet or console to the ace	4
CSCsh17936	show checkpoint all should include file size and timestamp	4
CSCsh36885	unable to clear the Denied counter for show resource usage	4
CSCsh56148	show probe invalid status code should display the actual code returned	4
CSCsh60110	snmp mib support needed for memory utilization	4
CSCsh60127	snmp mib support needed for resource usage	4
CSCsh86119	total current connections showing a vary large value	4
CSCsh96601	critical log message seen after configuring ft group	4
CSCsi21262	sticky cookie static can be configured to use wrong rserver	4
CSCsi31135	probe interval not working properly	4
CSCsi51462	ssl probe needs option to disable close-notify packet	4
CSCsi60550	cannot ping virtual IP from external dev when down and icmp-reply enable	4
CSCsi62078	removing limit-resources for sticky does not disable sticky	4
CSCsi62078	removing limit-resources for sticky does not disable sticky	4
CSCsi62321	ip verify reverse-path always enabled	4
CSCsi67832	high number of skipped probes for one rserver	4
CSCsj82350	Error: resources in use should be specific about the resource in questio	4
CSCsj82362	clear probe should have an argument for all	4
CSCsj84913	show ft group det sync status not in sync with standby	4
CSCsk38649	inactivity timer not closing connections when expected	4
CSCsh63341	removing specfic logging message leaves default in the running-config	5
CSCsi05345	leastconn slowstart documentation needs more detail	5
CSCsj00565	show ft group detail command has a difficult to read format	5
CSCsj47920	sysDescr OID isn't being correctly populated for the ACE	5
CSCsk33028	probe closes conn with reset when configured for graceful	5
CSCsg49360	no context deletes context without warning	enhancement (6)

Table L-3 DDTS of Interest but Not Encountered in DCAP 4.0 L2-3 ACE Testing (continued) (continued)

DDTS	Description	Severity
CSCsg49375	no context deletes checkpoints and saved files on disk0:	enhancement (6)
CSCsh12932	Save error logs in disk to check errors when in standby_cold	enhancement (6)
CSCsh81227	need show command detail information on failed connections	enhancement (6)
CSCsh86146	ssl traffic does not failover after running ft switchover	enhancement (6)
CSCsi51307	show probe detail does not display the learned and current hash	enhancement (6)
CSCsi60471	cannot ping virtual ip from local context	enhancement (6)
CSCsj93414	Cannot clear resource Usage Counters	enhancement (6)
CSCsk47599	Match-any does not work for both SSL and HTTP traffic	enhancement (6)

Volume 3: LAN (Layer 4-7) Services

The modular Catalyst 6500 switching platform supports various line cards which provide services at Layers 4-7, such as the Content Switching Module (CSM), Firewall Services Module (FWSM) and Application Control Engine (ACE). The tests in this chapter focus on the ability of these Service Modules to work together to provide load-balancing, and security services to data center traffic.

Two physically different deployments were tested in Cisco DCAP 4.0. In one, the Aggregation Layer switches are used to house Service Modules and to provide aggregation for the Access Layer. In the other, the Service Modules are deployed in separate Service Chassis that are connected to the Aggregation Layer switches. Testing was performed on each of these physically different topologies.

The following two Service Module combinations were tested in the Aggregation Layer switch deployment.

- Content Switching Module (CSM), Firewall Services Module (FWSM), Secure Socket Layer Services Module (SSLSM), and Intrusion Detection Services Module (IDSM)
- Application Control Engine (ACE), FWSM, and IDSM)

Though the CSM, FWSM, SSLSM, and IDSM combination was set up in the DCa topology (integrated in the Aggregation Layer switch) for many of these tests, for the majority of Cisco DCAP 4.0 testing, the ACE, FWSM, and IDSM combination was used in the Aggregation Layer switch. In the Service Chassis deployment, only the CSM, FWSM, SSLSM, and IDSM combination was tested.

In all of the various hardware configurations that were used in the testing, the Network Application Module (NAM) was installed and configured, though it wasn't tested directly at any time.

The following DDTS types were logged:

- L4-7 CSM DDTS Encountered, page L-14
- L4-7 CSM DDTS of Interest but Not Encountered, page L-14
- ACE DDTS Encountered, page L-20
- L4-7 ACE DDTS Encountered, page L-20
- L4-7 ACE DDTS of Interest but Not Encountered, page L-21
- L4-7 Service Switch (SS) DDTS Encountered, page L-28
- L4-7 IPS (IDSM) DDTS of Interest but Not Encountered, page L-28

L4-7 CSM DDTS Encountered

Table L-4 lists L4-7 CSM DDTS Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-4 DDTS Encountered in DCAP 4.0 L4-7 CSM Testing

DDTS	Descriptoin	Severity
CSCsl39483	show perfmon command does not show TCP intercept counters;	4

L4-7 CSM DDTS of Interest but Not Encountered

Table L-5 lists L4-7 CSM DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-5 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 CSM Testing

DDTS	Descriptoin	Severity
CSCek28276	SSLM:fdu core crashed on 1prob,256 proxy service creation	1
CSCec74017	STE:URL rewrite does not handle reloca string in multiple TCP pkts	2
CSCec84131	STE:SSLv2 forwarding, server connection gets struck in CLOSE_WAIT	2
CSCee46096	proxy service is down after deleting route and configue server vlan	2
CSCee54025	Locked RSA keys get overwritten when Cisco Router enrolled with CA	2
CSCee55894	SSM doesnt send GARP for its PHY int when HSRP is configured	2
CSCej01046	SSLM reset SSL connection from AON modules	2
CSCek01245	SSLM:Lower MTU(100) casues Frag SVC full drops- w/ big custom header	2
CSCek25662	STE: FDU should not crash when generating RST packet	2
CSCek31132	CSM L7 redirect causing SSLM to reboot	2
CSCek44398	CSM incorrectly setting TCP seq # on FTP PORT command to real	2
CSCek44961	FWSM: Doing show access-list would double count the ping traffic	2
CSCek67352	When doing clear mod x counter, the csm went offline & not revive	2
CSCek67352	When doing clear mod x counter, the csm went offline & not revive	2
CSCsa93383	TCL SSL probe between CSM and SSLM is failing	2
CSCsb84808	ssl sticky database not replicated after module is reset	2
CSCsc46105	Not carryover ToS value if enabling mls qos on Native IOS	2
CSCsd25820	SSL-M 2.1.8 keeps on rebooting in url-rewrite process	2
CSCsg58210	CSM forwards packets destined for real server out wrong vlan	2
CSCsh57876	CSM - FPGA core with 'service termination'	2
CSCsh63256	FWSM sets a CFI bit in dot1q header for traffic routed by FWSM	2
CSCsh89589	ARP fails on FWSM with SFM or SFM2 and S2/MSFC2	2
CSCsh98223	Redirect configuration can cause FPGA related core	2
CSCsi01910	Standby CSM not responding to sup keepalive polling - after config sync	2

Table L-5 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 CSM Testing (continued)

DDTS	Descriptoin	Severity
CSCsi13442	FWSM enabling ipv6 on interfaces causes ospfv2 adjacency down	2
CSCsi27367	Traceback in Thread Name: arp_forward_thread	2
CSCsi31953	FWSM crash at thread telnet , while running mgcp commands	2
CSCsi34576	FWSM problems copying config to/from a TFTP server	2
CSCsi40824	MTFW 3.1.4 Traffic loss on FWSM failover	2
CSCsi45384	FWSM - duplicate udp packet causes pc conn without a valid np conn	2
CSCsi50428	FWSM clearing the DSCP values when skinny inspection is enabled.	2
CSCsi51476	Denial-of-Service in VPNs with password expiry	2
CSCsi62117	FWSM crash Thread Name: fast_fixup	2
CSCsi63011	FWSM service-policy command causes high cpu (also possible failover)	2
CSCsi63421	FWSM crash Thread Name: doorbell_poll	2
CSCsi73738	High CPU due to ACK storm with inspect http	2
CSCsi86017	FWSM Memory leak with parser	2
CSCsi86017	FWSM Memory leak with parser	2
CSCsi90927	SIP inspection should not try to NAT and reject VIA 127.0.0.1 headers	2
CSCsi94230	SSH,Ping FWSM outside interface failure after failover	2
CSCsi95040	Connection-limits discrepancy between Syslog Message and show output	2
CSCeh91197	STE: Renewing CA cert is not used after reload	3
CSCeh97790	connections stuck in init state	3
CSCei08432	standby csm becomes unresponsive after config sync failure	3
CSCei45038	SSLM: PKI server initialization writting to NVRAM needs to change.	3
CSCei72718	New GET over persist conn, triggers CSM sending RST after maxconn reach	3
CSCej42499	CSM to fix SA parent warnings in casa/slb_laminar_chain.c	3
CSCek00631	conn counter on standby csm is incorrect	3
CSCek05878	sasp unique id overwritten when Config Sync is executed	3
CSCek08986	gslb probe shows operable but the serverfarm real is outofservice	3
CSCek21364	SSLM:strie_scan cannot alloc scan object : error on ssl console	3
CSCek24138	SSL:statistics for health probe feature is missing	3
CSCek25162	SSL:No warnning when route with next-hop ip same as server ip addres	3
CSCek25447	SSLM:H-probe should not probe for down services	3
CSCek25950	SSLM:health-probe timeout feature inconsistent for client proxy	3
CSCek27667	SSLM:fdu radix delete and add error on context removal and add	3
CSCek27840	fqdn changes to hostname when rsakeypair command is entered	3
CSCek28184	custom header length cannot exceed 239 char	3
CSCek28233	SSLM:Invalid status for tcp probe in case of 256 proxy,256 probes	3
CSCek28258	SSLM:TCP buffer leak after running traffic for days.	3

Table L-5 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 CSM Testing (continued)

DDTS	Descriptoin	Severity
CSCek28344	quotes are counted against the http-header char limit	3
CSCek28862	support needed for show context all	3
CSCek29235	total conn fail counter increments after SSL Alert w/Server RST	3
CSCek29512	custom header with spaces can be inserted without quotes in 3.1 code	3
CSCek29525	unable to remove header insert policy	3
CSCek29867	SSLM:clear ssl-proxy stats clear proxy running probe counter also	3
CSCek29977	SSLM:2nd probe change for proxy service not taking effect	3
CSCek30466	expired cert increments wrong counter during crl check	3
CSCek31281	when maxconn limit is reached no log message is generated	3
CSCek31283	when maxconn limit is reached new conns are not reset	3
CSCek31534	passphrase allows a short password	3
CSCek37109	CSM crashes w/ IXP3 SA-CORE exception due to mis-classified packets	3
CSCek37109	CSM crashes w/ IXP3 SA-CORE exception due to mis-classified packets	3
CSCek47179	header insert fails when using a large custom header	3
CSCek47179	header insert fails when using a large custom header	3
CSCek51742	csm takes 9+ seconds to arp for local host	3
CSCek51826	conn replication fails when reset master preempts	3
CSCek51826	conn replication fails when reset master preempts	3
CSCek51826	conn replication fails when reset master preempts	3
CSCek53290	status-tracking not making vserver outofservice	3
CSCek53290	status-tracking not making vserver outofservice	3
CSCek61080	Show mod csm x vser nam vs-name det shows zero server pkts	3
CSCek61080	Show mod csm x vser nam vs-name det shows zero server pkts	3
CSCek63293	CSM unexpected reload due to FPGA4 exception	3
CSCek64897	Core dump experienced	3
CSCek64897	Core dump experienced	3
CSCek67326	CSM drops ping to vserver when configured with service termination	3
CSCek68161	The maxconn seems to be one less than the actual one being shown	3
CSCek68456	Even non-existed SASP_SCALE_WEIGHTS still accept with no error	3
CSCek68612	Status-tracking shows operational even the vserver is down	3
CSCek68627	Status-tracking for a down vserver with bad probe shows operation	3
CSCek70084	When configuring any for virtual ip address, ping fails	3
CSCek70088	When configuring for tcp 4444 serv term on vip in vser,ping fails	3
CSCek70457	SASP_CSM_UNIQUE_ID variable disappears after config-sync	3
CSCek71183	When configuring a long insert header(63bytes),header insert fails	3
CSCek71982	When doing no vserver x, status-tracking of that vserver x disappear	3

Table L-5 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 CSM Testing (continued)

DDTS	Descriptoin	Severity
CSCek72343	Track mode all does not failover when down track interface(s)	3
CSCek72445	Removing preempt on active & standby csm doesnt change priority 254	3
CSCek72566	With heartbeat vlan down, both csms in intrachassis unavailable	3
CSCsb82667	STE: Bug to track health probe incomplete features	3
CSCsc45443	Delay in removing idle flows	3
CSCsd06143	show ssl-proxy stats hdr counter Service Errors is erroneously increment	3
CSCsd13447	CSM incorrectly set GSLB real state to down after reload	3
CSCsd24461	Configuring CSM with SSL stickyness shows as src-ip stickyness.	3
CSCsd27478	FPGA4 exception w/ icp.fatPath length error (icpFatErr)	3
CSCsd62698	CSM: Overflow Errors on Standby CSM result in connection leaks	3
CSCsd63761	traffic cannot pass through FWSM in transparent mode	3
CSCsd69971	GSLB does not fail real and probe when configured with wrong address.	3
CSCsd70038	GSLB responds with a DNS requested timed out message."	3
CSCsd80681	CSM: Config-sync of large configs sometimes causes reload	3
CSCsd98783	CSM SASP dfp agent still reporting real after being removed from SF	3
CSCsd98833	CSM SASP New real is up and operational with a weight of zero.	3
CSCsd99863	CSM uses predictable TCP Initial Sequence Numbers when doing L5/L7 LB	3
CSCse04645	CSM L7 with cookie-insert causes many out-of-sequence packet	3
CSCse06717	no Lam Telnet task so CSM impossible to session into CSM	3
CSCse14356	Its not possible to clear single conn with clear conn on 3.1.1	3
CSCse21474	CSM show mod csm X conns command list RTSP data channel in INIT state	3
CSCse53716	Client RSTs or Server SYN-ACKs with data can cause buffer loss	3
CSCse76730	CSM:Core with header FPGA3 IXIC_TAGERR	3
CSCse97370	Connection Replication across redundant CSMs fails.	3
CSCsf11008	CSM allows global real to be created and put inservice without address	3
CSCsf19815	Vserver out-of-service when primary real fails over to backup real	3
CSCsf21551	HTTP probe fails when server sends 200ok & then resets the connection	3
CSCsg11521	CSM: change XML socket to listen only on 1 ip instead of 0.0.0.0	3
CSCsg11531	CSM does not update active connections when destination mac address flap	3
CSCsg20504	CSM - Status-tracking can cause deadlock, % No ICC response for TLV	3
CSCsg40777	Cannot reach CSM VIP when NATd at sup720 interface	3
CSCsg40988	CSM crash with FPGA3 exception : internal data with a zero buffer length	3
CSCsg45612	CSM stops processing requests IXP1 exceeds 100% utilization	3
CSCsg51792	small/medium buffer leak (IXP2) with service termination	3
CSCsg59530	unable to connect to vserver via telnet on a csm.	3
CSCsg72976	CSM - need to add standby state to mib object slbRealServerState	3

Table L-5 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 CSM Testing (continued)

DDTS	Descriptoin	Severity
CSCsg73312	CSM is bridging Client and server Vlan for different subnet	3
CSCsg84530	CSM may unexpectedly reload with PPC exception type 1792 in FTReplFlow	3
CSCsg91075	CSM core in SASPComm due to corrupted FD structure	3
CSCsg94630	sticky issue when CSM up for longer than 497 days	3
CSCsh22019	FWSM 3.1.4 ACL hitcnt not incrementing for authenticated traffic	3
CSCsh22071	FWSM 3.1.4 show np 3 aaa stats counter AAA Lookup Failures Incrementing	3
CSCsh25401	Config-sync breaks after copying startup-config to running config	3
CSCsh32023	FWSM accepts wildcard mask for ip local pool	3
CSCsh35930	ESMTP inspection drops emails with special characters in the email addr	3
CSCsh41522	Logging queue 0 is wrongly shown as being unlimited queue	3
CSCsh43381	Partial Serverfarm Failover not working	3
CSCsh46620	CSM processes ARP entries from incorrect vlans	3
CSCsh46799	Add config parm to permit network address of static to be used	3
CSCsh49717	Need to add IP land attack checks to ingress packet processing	3
CSCsh51650	acl hitcnt for policy nat is doubled	3
CSCsh53633	CSM 4.2(6) - L7 abort crash seen with empty egress FIFO	3
CSCsh64882	Adding new status tracking vserver doesn't remove old one	3
CSCsh67596	TFW: Arp flowing across the box is not updating the bridge-timeout	3
CSCsh68423	Traceback in Thread Name: fast_fixup due to SUNRPC inspection	3
CSCsh71726	Removing tracked status tracking vserver doesn't remove dep. association	3
CSCsh74881	CSM with a pair of bridged vlans can cause a variable to not function	3
CSCsh75142	Cannot rename interface after remove/add its VLAN to context	3
CSCsh77510	CSM core in FPGA3 while processing an arp response from gateway	3
CSCsh83504	CSM generates conflicting cookie hashes instead of unique ones	3
CSCsh84334	ssh process never terminates if aaa server does not respond	3
CSCsh90052	Error message missing, if failover activation is failing	3
CSCsh90755	CSM may not insert cookie if Connection: close header has in response	3
CSCsh92908	failover device deletes static routes from configuration	3
CSCsh96686	SASP task hung	3
CSCsh97689	Upgrade from 2.x - 3.x allocate-interface map_ifc string is lost	3
CSCsh98336	FWSM 3.1.4 MRFW Radius Downloadable ACL hitcnt not incrementing properly	3
CSCsi01822	ACL might not block packet if nat-control is disabled	3
CSCsi05248	ICMP unreachable messages fail through FWSM with inspect icmp error	3
CSCsi10418	Gratuitous ARP support in FWSM classifier	3
CSCsi12105	mac-addresses found on failover vlans generate static table entries	3
CSCsi12289	FWSM Does Not Display Correct Timezone for DST	3

Table L-5 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 CSM Testing (continued)

DDTS	Descriptoin	Severity
CSCsi15190	FTP PORT command not processed when TCP Normalizer is active	3
CSCsi18503	Slow memory leak in H323 RAS inspection	3
CSCsi23644	CSM crash with FPGA2 exception: iPacket and ePacket tags don't match	3
CSCsi24332	FWSM - slow memory leak when filter ftp enabled	3
CSCsi24354	CSM core dump IXP2 SA-CORE (Ex 5)(00000000h)	3
CSCsi27512	FTP with multiline 221 lines closes the connection too early	3
CSCsi28206	FWSM multictx transp. failover - xlates on standby affects failover	3
CSCsi28487	FWSM 3.1.4 pim rp-address interface command is not showing up in running	3
CSCsi34866	FWSM logging rate-limit num value of 1 causes command to remain hidden	3
CSCsi40555	FWSM service-policy global_policy global missing by default	3
CSCsi40571	FWSM can't apply more than one inspection engine to a class	3
CSCsi40607	FWSM icmp error inspection engine doesn't work	3
CSCsi42756	FWSM memory leak when adding and removing context	3
CSCsi43659	DOC: Active FTP is not blocked with filter ftp command enabled	3
CSCsi45294	FWM: Some PAT xlates do not timeout after 30 secs	3
CSCsi48952	Multicast traffic builds invalid xlate causing traffic interruption	3
CSCsi50214	FWSM is unable to use VLAN 4094	3
CSCsi54090	FWSM: traceback when removing capture	3
CSCsi58010	FWSM:LU allocate xlate failed on standby unit	3
CSCsi63925	Auth proxy generates login form with server IP address instead of name	3
CSCsi68641	show route stat causes ERROR: np_logger_query request	3
CSCsi73723	106101 syslog is sent while total amount of deny flows is lower than max	3
CSCsi75805	FWSM transparent mode allows telnet to outside interface	3
CSCsi79428	FWSM 3.1.4 may reload when viewing running configuration.	3
CSCsi87893	Strange output may appear while configuing tftp-server parameters	3
CSCsi92378	Fover: Primary Standby sends GARP with Active MAC when start-up	3
CSCsi96676	Local authentication password fails following upgrade.	3
CSCsi97028	FWSM answers for Ascii FTP connection request stating checksum is bad	3
CSCsj00178	Unable to Access FWSM Over SSH or ASDM	3
CSCsj00183	FWSM MR mode error message when coping capture file from FWSM to Linux T	3
CSCsj04323	non ftp/telnet/htttp traffic triggers authentication process	3
CSCsj07948	management-access inside causes crash when FWSM brings up L2L Tunnel	3
CSCsj09074	FWSM: norandomseq does not work with NAT exemption	3
CSCeh81581	sh crypto ca trustpoints does not display server cert or config	4
CSCek01699	csm dfp agent config syntax is incorrect	4
CSCek08685	csm sends dns answer for non-existent domains to ns forwarder	4

Table L-5 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 CSM Testing (continued)

DDTS	Descriptoin	Severity
CSCek28201	when max length for custom header is exceeded vague log message	4
CSCek28746	SSLM:IOS Core s/w reset after fdu core crashed	4
CSCek31276	clear ssl-proxy stats context does not clear maximum conns counter	4
CSCek31540	RSA key lock uses first 16 bytes of passphrase - display warning	4
CSCek50556	No ICC response for TLV type 478 from CSM linecard	4
CSCek59900	Configuring virtual tcp service termination fails ping to vip	4
CSCsh66769	FWSM: hitcnt for some ACLs still see the counter doubled	4
CSCek28145	sslm system clock behavior has changed from version 2.1.x	5
CSCek28849	valid session counter is a misleading description	5
CSCek01033	csm does not set member state when connecting to sasp GWM	enhancement (6)
CSCse97898	FWSM need a generic clear access-list counters command	enhancement (6)
CSCsi32262	fwsm should recover from a stuck np3	enhancement (6)

ACE DDTS Encountered

Table L-6 lists ACE DDTS Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-6 DDTS Encountered in DCAP 4.0 ACE Testing

DDTS	Descriptoin	Severity
CSCsj30713	ping to some of the vips fails after policy-map config change	3
CSCsj66157	TCP Reset from client not processed	3
CSCsj85316	ICMP Err Inspection: Translate Real IP to VIP in IP header of ICMP pkt	3
CSCsj89960	ACE - generates error when using global NAT address with diff ports	3
CSCsj99175	Deletion failed for Sfarm Probe Table	3
CSCsk00341	Unconfigured static cookie shown in sh run & still provides stickness	3
CSCsk01206	SSH key generation message and no keys generated	3
CSCsk10290	Different interface Id's for Active/Standby cause xlate error.	3

L4-7 ACE DDTS Encountered

Table L-7 lists L4-7 ACE DDTS Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-7 DDTS Encountered in DCAP 4.0 L4-7 ACE Testing

DDTS	Descriptoin	Severity
CSCsg62851	c6k-ace:lb:'cookie insert' inserting cookies w/ zero-length values;	2
CSCsh14278	sh serverfarm failure conn incremented for successful connection;	3
CSCsl50509	ace does not pass fin from client or server;	3
CSCsl66036	incorrect output for http insepct;	4

L4-7 ACE DDTS of Interest but Not Encountered

Table L-8 lists L4-7 ACE DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-8 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 ACE Testing

DDTS	Descriptoin	Severity
CSCsi44096	feature CONN-REUSE not functioning without failed sessions	1
CSCsj38511	HM:while running scripted probes test case 11.2.1.3, ACE get a core	1
CSCsj38645	SSL init stress using RC4-MD5 crashes ACE	1
CSCsj55836	Buffer leak / crash with FE-SSl traffic with L7 features configured	1
CSCsk30865	X_TO_ME: Hung while running FE with client auth with 4k bits key	1
CSCsk35523	cfgmgr crash during config/unconfig of 250 contexts and ft groups	1
CSCsk48871	Kernel Crash on trying on doing a tftp copy soon after bootup	1
CSCsk49756	ACE NP Crash with FE-SSL trafific having Match-src as class	1
CSCsk50946	ssl is crashed when applying a chaingroup with 8 certificates	1
CSCsf09910	ACE Conn Repl: FTP Data channel stalls after swtichover with inspection	2
CSCsh19664	scripted probe uses all resources and then causes all probes to fail	2
CSCsi36216	traffic will not go to one particular rserver	2
CSCsi69018	sticky database not in sync after ft switchover with ACE in bridge mode	2
CSCsi70738	config will not sync running 2 ACEs in same chassis in bridge mode	2
CSCsi79974	Syslog not sending CP or DP level 6 messages.	2
CSCsj12610	xml script doing 'show version' on ACE causes silent reload after 30 hrs	2
CSCsj14813	SSL connections fail after changing the certs and Keys with traffic	2
CSCsj16667	HM crash when rserver put inservice after long period of being OOS	2
CSCsj17220	L4Lb with IP sticky and NAT under stress causes many ME's to be stuck	2
CSCsj24815	Standby ACE crashed during A1(4n) to A1(5) throttle image upgrade	2
CSCsj25440	ACE/Scimitar:no radius/tacacs+/ldap failed	2
CSCsj25941	ACE - ACE reboot w/core NP 0 Failed : NP ME Hung during Fragmenting	2
CSCsj26023	QNX qconn process core	2
CSCsj37029	A15: Telia SSL performance test case does not pass	2

Table L-8 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 ACE Testing (continued)

DDTS	Descriptoin	Severity
CSCsj37202	buffer leaked after running front-end ssl traffic for 24 hours	2
CSCsj37653	HTTP: Wrong deobfuscation of multibyte UTF8 encoding in URL	2
CSCsj56916	L7 Connectings failing with LB policy misses	2
CSCsj59659	ACE crash with Mutiple ME's stuck with L7 Lb traffic	2
CSCsj59659	ACE crash with Mutiple ME's stuck with L7 Lb traffic	2
CSCsj74391	ACE does not g-ARP NAT Address upon reload	2
CSCsj77194	Intermittent erroneous XML response for show context in Admin context.	2
CSCsj88500	<show context> & <show service-policy> display incomplete/truncated info	2
CSCsj95609	TCP: TCB and other TCB has same reassembleq	2
CSCsk03039	malformed xml returned from show_context	2
CSCsk07539	Stress:Multiple ME stuck with SSL-INIT traffic havig multiple L7 Feature	2
CSCsk26606	ACE: malformed SSL record if certificate chain length higher than 4k	2
CSCsk30079	HA flapping while running FE SSL with client auth perf test	2
CSCsk30714	internal buffers leaked after running FE ssl with client-auth	2
CSCsk33251	ACE: IP Source stickiness can fail after upgrade and/or failover	2
CSCsk35058	Initial TCP sessions are dropped after a reboot	2
CSCsk35629	unable to login. Internal CLI error: No such file or directory	2
CSCsk36611	ACE: ssl re-handshake fails with IE when cert chain bigger than 4k byte	2
CSCsk44718	ace core dump after sup sso failover	2
CSCsk50407	ACE breaks application by performing implicit PAT on loadbalanced conns	2
CSCse70005	tcp port numbers not converted correctly with csm2ace tool	3
CSCse70052	csm2ace tool: maxconns not converted as conn-limit in serverfarm	3
CSCse70823	csm2ace tool: does not convert serverfarm predictor	3
CSCse70865	csm2ace tool: serverfarm real probe not applied to ACE config	3
CSCse72427	csm2ace tool: places serverfarm probe in wrong location	3
CSCse90603	auto complete doesn't work on checkpoint names	3
CSCsg06046	persistent-rebal causes all persistent gets to be remapped on the server	3
CSCsg06046	persistent-rebal causes all persistent gets to be remapped on the server	3
CSCsg10024	ftp L7 denies cdup but not cd ..	3
CSCsg23016	class map insert-before doesn't work on previously configured class map	3
CSCsg23096	show conn serverfarm not showing ftp data connection	3
CSCsg52355	RHI Injected routes lost after SUP switchover	3
CSCsg70663	csm2ace tool - csm config should be ported to ACE usr context, not Admin	3
CSCsg70682	csm2ace tool - nat not supported	3
CSCsg70686	csm2ace tool - naming convention for resource class + undefined objects	3
CSCsg70689	csm2ace tool - no alert to the user of a script not being present on ace	3

Table L-8 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 ACE Testing (continued)

DDTS	Descriptoin	Severity
CSCsg70696	csm2ace tool - converstion of multi-policy to ace is not correct	3
CSCsg70702	csm2ace tool - class-map inconsistency	3
CSCsg70709	csm2ace tool - loadbalance vip icmp-reply active (needs to be added)	3
CSCsg70719	csm2ace tool - inconsistent naming convention (case sensitive)	3
CSCsg70731	csm2ace tool - sticky config on csm only partically converts to ace	3
CSCsg70736	csm2ace tool - command <replicate sticky> NOT <replicate-sticky> on ace	3
CSCsg70742	csm2ace tool - description feild (vlan) not converted fom csm to ace	3
CSCsg71636	csm2ace tool - incorrect cli command for ace class-map (dns vs domain)	3
CSCsg71838	csm2ace tool - invalid regular expression (conversion from csm to ace)	3
CSCsg73137	csm2ace tool - csm port on probe is inherited from the vservers not ACE	3
CSCsg73181	cms2ace tool - ACE conversion via CSM omits last for regex string	3
CSCsg73521	csm2ace tool - static arp entry on csm not listed in unsuported commands	3
CSCsg73557	csm2ace tool - capp udp optoins unsupported comand on csm omitted	3
CSCsg78690	sh tech returns errs for all CLIs that the user's role denies access to	3
CSCsg80474	SSL show stats commands missing - only ssl stats are ref in ucdump	3
CSCsg80625	spanned udp request (not frag) packets are dropped by ace	3
CSCsg80625	spanned udp request (not frag) packets are dropped by ace	3
CSCsg86626	after upgrade failure unable to boot prior image from rommon	3
CSCsg88014	copy ftp image fails when overwriting the same filename	3
CSCsg91127	license mismatch after member is added to context	3
CSCsg98014	1(Error: resources in use, attempting to add a RC to a context	3
CSCsh06784	Ingress vlan SPAN send all vlan to destination port	3
CSCsh20546	maxconn, persistence and sticky not working together	3
CSCsh27746	tacacs not working with Cisco Secure ACS	3
CSCsh53696	resource usage show current value greater than max allowed	3
CSCsh53739	rserver shows arp_failed, but probe, arp and access exists	3
CSCsh72916	poor performance with persistent rebalance and url hash load balancing	3
CSCsh81195	total conn-failures incrementing when client sends reset on estab conn	3
CSCsh82790	client request delay if server mtu is set to 250	3
CSCsh91414	rhi routes not injected or removed quickly after forced ft failover	3
CSCsi13650	persistent HTTP 1.1 conn that reaches max-conn limit gets reset	3
CSCsi20428	current conn counter on serverfarm shows active conn with no flows	3
CSCsi23858	probe failure when changing config	3
CSCsi26014	counters not incrementting when exceeding parse length	3
CSCsi43733	scripted tftp probe shows failures	3
CSCsi43733	scripted tftp probe shows failures	3

Table L-8 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 ACE Testing (continued)

DDTS	Descriptoin	Severity
CSCsi52002	sticky warning log message after reload	3
CSCsi60741	Error 12459: :[12459] Login failed	3
CSCsi62289	ace does not decrement the ttl when packets route through it	3
CSCsi80046	tac-pac cmd issues	3
CSCsi81930	PSS error while configuring snmp-server community in client context	3
CSCsi82962	Certain incoming packet data causes ACE to lose MSS setting mid-flow.	3
CSCsi88102	mts error mts_acquire_q_space() failing - no space in sap 2914	3
CSCsi92341	ACE sh logging may not show latest log	3
CSCsi93631	ACE capture feature may disregard bufsize limit.	3
CSCsi98829	SSL handshake fails using self signed certificate	3
CSCsj04439	Configuring no logging message 313004 shows up in config twice.	3
CSCsj04447	domain checks for scripted probes are missing	3
CSCsj04464	domain checking for monitoring sticky database not working	3
CSCsj07054	capture command with circular-buffer option not overwriting the buffer	3
CSCsj08692	Stanby Blade crashed on reloading Active with L3/L7/SSL traffic	3
CSCsj13557	Sticky database entries of active connections disappear after HA upgrade	3
CSCsj15501	'aaa' config in Admin context shows up in user context	3
CSCsj20156	ACE tac-pac generation may fail with `No space left on device` message.	3
CSCsj20171	Cannot remove 'service policy input global'	3
CSCsj21032	Improve performance of crb_compute_diff()	3
CSCsj21178	itasca_ssl(911) has terminated on receiving signal 11, key import	3
CSCsj23996	acl download failure due to duplicate lineno	3
CSCsj24580	user ft group stuck in bulk_sync because of missing arp	3
CSCsj25054	CMCLOSE and TCP stuck with Pipelined traffic and rebalance	3
CSCsj26329	SNMP walk shows low out packet for Baikal compared to the device itself	3
CSCsj27746	golden: ftp data transfer hang after ft switchover	3
CSCsj27882	Golden: ha_basic fails in test 5.10. FT group in STANDBY_COLD	3
CSCsj28335	Poor throughput with L7LB and persistence rebalance	3
CSCsj30048	ACE:ssh key is not synced to STANDBY	3
CSCsj31419	ACE reset by SUP due to keepalive polling failure	3
CSCsj34419	Replicate connection if xlate error counter is increase	3
CSCsj35046	checkpoint rollback issues with banner and hostname/ssh keys	3
CSCsj37212	crashinfo created	3
CSCsj40904	tcp unproxy can fail with unproxy_ack w/fin flag set	3
CSCsj41909	ACE: NP 1 Failed : NP Process: QNX process io_net Crashed	3
CSCsj43225	ACE: Sticky feature issues after downgrading from A1(5a) to A1(4n)	3

Table L-8 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 ACE Testing (continued)

DDTS	Descriptoin	Severity
CSCsj47861	Demo license install failure	3
CSCsj48884	ACE: ftp data channel fixup must have priority over policy nat	3
CSCsj49249	server info isn't updated while learning the same cookie from other srv	3
CSCsj51148	Elapsed time is not taken into account while changing the sticky timer	3
CSCsj51161	NAT policy in DP not updated if nat pool removed	3
CSCsj51637	xml agent should do roll back when error occur	3
CSCsj52843	Syslog for rserver state change due to the probe failure not generated	3
CSCsj53114	rserver's sticky is removed while rejecting the conns. due to MAXCONNS	3
CSCsj55058	Existing connection is not closed while changing the sticky serverfarm	3
CSCsj58598	ME core with ICM util at 100%	3
CSCsj61121	ACE is not learning the multiple instances of the same cookie	3
CSCsj62850	Disabling timeout activeconns still flushes active conns. sticky entry	3
CSCsj63564	HA: auto-sync not cleared when context ID is reused.	3
CSCsj64177	sfarm total conn counter is updated for each cookie request in same conn	3
CSCsj64723	ACE: http/https probe no hash does not clear the reference hash value	3
CSCsj64822	ace needs process to track and provide supplied probe script updates	3
CSCsj64833	UDP/TCP traceroute does not work to configured ACE IP Interfaces	3
CSCsj65569	ACE: LB Policy miss with match souce-address clause in L7LB policy	3
CSCsj65699	No Syslogs are getting generated while Disabling probe manually	3
CSCsj68643	can_wait_specific_msg: Aborting call (SAP 27, pid 959)	3
CSCsj70522	log messages not being displayed on console or in buffer	3
CSCsj70624	TCP reset during HTTP traffic	3
CSCsj74004	Issue with adding nat dynamic and nat-pool to an empty service policy	3
CSCsj74250	Key not encrypted when Configuring TACACS on ACE	3
CSCsj74292	ACE Subcontext Configuration Lost with FT Config & No Peer	3
CSCsj74518	NAT dnld err when removing a nat-pool that has multiple IP ranges	3
CSCsj80265	SSHv1 with TACACS sessions into ace fail	3
CSCsj84841	HTTP: Improve reproxy performance of chunked response	3
CSCsj90095	ooo out-of-order packets in response sent from ace to client	3
CSCsj91529	Incorrect syslog is generated while deleting the class map.	3
CSCsj91836	Trinity: RTSP Inspect adjusting content-length of SDP msg incorrectly	3
CSCsj93358	VIP has to be in inservice even though all rservers are in MAXCONNS stat	3
CSCsj94366	unable to change console settings	3
CSCsj95752	MAXCONNS rserver's total conn counter is updated for each request	3
CSCsj97786	Removing service-policy from one vlan deletes from all	3
CSCsj99704	Regex resource clas minimums not guaranteed in contexts	3

Table L-8 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 ACE Testing (continued)

DDTS	Descriptoin	Severity
CSCsk02229	Ephemeral step up for SSL initiation does not work	3
CSCsk03514	ACE crashes during normal operation	3
CSCsk08296	Changing L7 policy disrupts existing traffic flows	3
CSCsk10045	current conn. counter is not all decremented with server-conn reuse	3
CSCsk15979	idle connection closed before timeout	3
CSCsk16083	https probe using hash always show success with Windows 2000	3
CSCsk16823	syslog buffer problems when increasing resource limit	3
CSCsk19394	static sticky entries aren't generated for rservers in backup serverfarm	3
CSCsk21143	Connections lost with ft preempt and heartbeat lower than 200ms	3
CSCsk23763	Global tacacs key configuration doesn't work	3
CSCsk25291	conns. are not load-balanced to the rserver returned from MAXCONNS state	3
CSCsk25567	Uninstall contexts license should check the numbers of contexts	3
CSCsk26767	ACE: can't delete or modify chaingroup even if not being used	3
CSCsk28161	sh tech o/p partially xmlized	3
CSCsk31782	high CPU utilization running large number of probes	3
CSCsk32367	backup rserver's weight should not be included for weight normalization	3
CSCsk33385	Spontaneous Crash with Fastpath Stuck	3
CSCsk33613	can't rollback from load-balance policy-map to http load-balance policy	3
CSCsk34767	unable to clear ftp inspect stats	3
CSCsk34831	ACE HealthMon (HM) corefile cannot be loaded in gdb	3
CSCsk34843	Data Bus error messages on the console	3
CSCsk34907	incremental sync failure but config is in sync	3
CSCsk34973	ftp inspect policies remain after removing class-map from policy-map	3
CSCsk36590	ACE: QNX core in LbUtil_ServerGoodListCheck	3
CSCsk37369	Error in Message receive : Interrupted function call	3
CSCsk38667	ACE fails to send window probe to host sending window 0 after conn. estb	3
CSCsk39152	ACE: cat: write error: No space left on device	3
CSCsk42541	Error: Called API encountered error	3
CSCsk42839	sticky resource not discovered for this context -- type 85	3
CSCsk43066	HA state STUCK in TL_SETUP, after upgrading to A161	3
CSCsk43321	system crash while generating an IXP crash corefile.	3
CSCsk46504	ACE experienced an ICM microengine core during normal operation A1(4L)	3
CSCsk46544	logging monitor is not synched to standby	3
CSCsk46771	keys with file name more then 64 characetrs messed up the sh crypto comm	3
CSCsk49388	csr with 2048 bit key is generated with some extra characters	3
CSCsk49666	RTSP Connection hangs after switchover with inspection and Nat	3

Table L-8 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 ACE Testing (continued)

DDTS	Descriptoin	Severity
CSCsk51091	HTTP Error Code Monitoring breaks https to WAAS Farm	3
CSCsk51599	check-point rollback is not clearing interface vlan config	3
CSCse90570	unable to break out of copy command with ctrl-c	4
CSCsg07468	service-policy packet counter wrong when using tcp-reuse	4
CSCsg07544	service-policy hit counter is misleading	4
CSCsg07570	show service-policy does not show advanced-options that are configured	4
CSCsg23578	when clearing stats the command stats should be used	4
CSCsg24832	quickly adding and removing ftp inspect policy cause policy error messag	4
CSCsg42659	1 connection counts as 2 on various show commands	4
CSCsg58753	show sticky database has a difficult to read format	4
CSCsg58769	show sticky database lacks important detail	4
CSCsg58783	show sticky database or with group does not display static entries	4
CSCsg75273	show conn serverfarm does not have detail argument	4
CSCsg75308	show serverfarm <name> detail should include some probe information	4
CSCsg88045	copying file to image transfer mode is not enforced	4
CSCsg92839	ssl probe - unable to select 5 ciphers	4
CSCsg93314	unable to delete an rsa key when name is >= to 86 char.	4
CSCsh12734	ACE-1-727005: HA: Configuration Replication failed needs more detail	4
CSCsh12819	no log message when ft auto-sync executed on standby	4
CSCsh13030	unable to paste a small config through telnet or console to the ace	4
CSCsh17936	show checkpoint all should include file size and timestamp	4
CSCsh36885	unable to clear the Denied counter for show resource usage	4
CSCsh56148	show probe invalid status code should display the actual code returned	4
CSCsh60110	snmp mib support needed for memory utilization	4
CSCsh60127	snmp mib support needed for resource usage	4
CSCsh86119	total current connections showing a vary large value	4
CSCsh96601	critical log message seen after configuring ft group	4
CSCsi21262	sticky cookie static can be configured to use wrong rserver	4
CSCsi31135	probe interval not working properly	4
CSCsi51462	ssl probe needs option to disable close-notify packet	4
CSCsi60550	cannot ping virtual IP from external dev when down and icmp-reply enable	4
CSCsi62078	removing limit-resources for sticky does not disable sticky	4
CSCsi62078	removing limit-resources for sticky does not disable sticky	4
CSCsi62321	ip verify reverse-path always enabled	4
CSCsi67832	high number of skipped probes for one rserver	4
CSCsj82350	Error: resources in use should be specific about the resource in questio	4

Table L-8 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 ACE Testing (continued)

DDTS	Descriptoin	Severity
CSCsj82362	clear probe should have an argument for all	4
CSCsj84913	show ft group det sync status not in sync with standby	4
CSCsk38649	inactivity timer not closing connections when expected	4
CSCsh63341	removing specfic logging message leaves default in the running-config	5
CSCsi05345	leastconn slowstart documentation needs more detail	5
CSCsj00565	show ft group detail command has a difficult to read format	5
CSCsj47920	sysDescr OID isn't being correctly populated for the ACE	5
CSCsk33028	probe closes conn with reset when configured for graceful	5
CSCsg49360	no context deletes context without warning	enhancement (6)
CSCsg49375	no context deletes checkpoints and saved files on disk0:	enhancement (6)
CSCsh12932	Save error logs in disk to check errors when in standby_cold	enhancement (6)
CSCsh81227	need show command detail information on failed connections	enhancement (6)
CSCsh86146	ssl traffic does not failover after running ft switchover	enhancement (6)
CSCsi51307	show probe detail does not display the learned and current hash	enhancement (6)
CSCsi60471	cannot ping virtual ip from local context	enhancement (6)
CSCsj93414	Cannot clear resource Usage Counters	enhancement (6)
CSCsk47599	Match-any does not work for both SSL and HTTP traffic	enhancement (6)

L4-7 Service Switch (SS) DDTS Encountered

Table L-8 lists L4-7 Service Switch (SS) DDTS Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-9 DDTS Encountered in DCAP 4.0 L4-7 SS Testing

DDTS	Descriptoin	Severity
CSCec74017	STE:URL rewrite does not handle reloca string in multiple TCP pkts;	2
CSCsj16292	DATACORRUPTION-1-DATAINCONSISTENCY: copy error .;	3
CSCsl39483	show perfmon command does not show TCP intercept counters;	4
CSCeh70549	clear ssl-proxy stats does not clear all counters;	5

L4-7 IPS (IDSM) DDTS of Interest but Not Encountered

Table L-10 lists L4-7 IPS (IDSM) DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-10 DDTS of Interest but Not Encountered in DCAP 4.0 L4-7 IPS (IDSM) Testing

DDTS	Descriptoin	Severity
CSCsl55330	Spanning tree state not OK for IDSM data port	2
CSCsm04146	IDSM2: sensor is not learning host OS when configured in Inline mode	2
CSCsd19619	NO statistics on traffic under heavy load	3
CSCsi33423	IDSM:intrusion-detection mod data-port capture allowed-vlan rejected if	3
CSCsb77175	CPU exceeds 100%	5
CSCsb74402	FTP Timeout not being applied	enhancement (6)

Volume 4: Storage Area Networking (SAN)

The DCAP SAN topology incorporates Cisco MDS fabric director products and design guides, industry best practices, and storage vendor implementation guidelines to provide a SAN infrastructure that is representative of the typical enterprise data center environment. The centerpiece of the topology is the Cisco MDS 9513 multi protocol SAN director running SAN-OS version 3.1(3a). The Cisco MDS 9124e embedded SAN fabric switch is also part of the topology.

The topology provides redundant fibre channel connectivity for Linux and Windows hosts using QLogic and Emulex host bus adaptors (HBA) to three different types of fibre channel enterprise storage arrays, namely the EMC DMX3, NetApp FAS6070, and Hewlett Packard XP10000. The topology also provides redundant fibre channel connectivity for synchronous storage replication and fibre channel over IP (FCIP) connectivity for asynchronous storage replication. Delay simulators and cable spools allow modeling of a redundant data center environment for disaster recovery and business continuance testing. The topology is designed to use actual hosts and applications to generate test traffic to model actual customer environments as close as possible.

The topology also includes a Quantum (formerly ADIC) i500 Scalar tape library with two IBM LTO3 tape drives.

The following DDTS types were logged:

- SAN DDTS Filed, page L-29
- SAN DDTS of Interest but Not Encountered, page L-30

SAN DDTS Filed

Table L-11 lists SAN DDTS Filed by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-11 DDTS Filed in DCAP 4.0 SAN Testing

DDTS	Descriptoin	Severity
CSCsk55538	Fabric Manager does not remove empty VSANs after adding members.;	4
CSCsk96269	Sup2 (slot 8) removed leaves a Fan Status note on blank module in DM;	4

SAN DDTS of Interest but Not Encountered

Table L-12 lists SAN DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-12 DDTS of Interest but Not Encountered in DCAP 4.0 SAN Testing

DDTS	Descriptoin	Severity
CSCsj49207	FCIP TA link flap and core due to Host asking for status a second time.	1
CSCsg19303	Scalability: packet drop with port-channel flapping	2
CSCsg49151	links do not get isolated on merge when SDV domain clashes with real dom	2
CSCsi49231	Switch CPU 100% constantly because of FLoGI storm	2
CSCsi72048	9216i FCIP link fails due to Heartbeat failure in encryption engine	2
CSCsj72662	MDS gen 1 linecards fail on upgrade with Q-Engine failure VOQMEM parity	2
CSCsk21652	fcip flaps when target sends error status without conf bit set (tlb)	2
CSCsk22374	PIO read parity errors on ASIC TCAM (egress) during upgd from 3.1(2b).	2
CSCsi56949	switch assigns same domain id when vsan created at same time on each sw	3
CSCsj29134	Upgrade to 3.1.2, ports are not coming online	3
CSCsj50299	INCIPIENT: ssh ver1 cores, upgrade fails with Return code 0x4093001E	3
CSCsj52389	snmpd - mem-leak in libzscmi	3
CSCsj14140	Alarms/condition retrieved with incorrect aid	4
CSCsj44453	ivr rewrite for device not successful (in rscn_offline_sent state)	4

Volume 5: Wide Area Application Services (WAAS)

Cisco Wide Area Application Services (WAAS) is an application acceleration and WAN optimization solution for geographically separated sites that improves the performance of any TCP-based application operating across a wide area network (WAN) environment. With Cisco WAAS, enterprises can consolidate costly branch office servers and storage into centrally managed data centers, while still offering LAN-like service levels for remote users.

The DCAP WAAS topology incorporates the Wide-area Application Engines (WAE) at the remote branch and in the data center, either at the DC WAN edge or at the aggregation layer. For TCP traffic redirection at the WAN edge of Data Center B, Web Cache Communication Protocol version 2 (WCCPv2) was used. At Data Center A the Cisco Application Control Engine (ACE) was used at the data center aggregation layer for transparent TCP redirection. The tests in this chapter focus on the functionality of the WAAS software on the WAE devices as well as the ability of the data center ACE and WAN Edge routers to intercept and redirect TCP-based traffic. Microsoft Exchange 2003 and Oracle 11i E-Business Suite traffic was sent and optimization and functionality were verified and quantified.

The following DDTS types were logged:

- WAAS ACE DDTS of Interest but Not Encountered, page L-31
- WAAS WCCP DDTS Encountered, page L-32
- WAAS WCCP DDTS of Interest but Not Encountered, page L-32
- WAAS WCCP DDTS Previously Encountered Not Fixed, page L-33

WAAS ACE DDTS of Interest but Not Encountered

Table L-13 lists WAAS ACE DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-13 DDTS of Interest but Not Encountered in DCAP 4.0 WAAS ACE Testing

DDTS	Descriptoin	Severity
CSCsg11506	EPM breaks connections if it doesn't intercept both traffic directions	2
CSCsi44131	Wrong WAFS core selected with CIFS AD	2
CSCsk34237	Egress multicast replication broken due to wccp .	2
CSCsf02614	WAFS misses keepalive (PE error code 246)	3
CSCsh93105	Changing MSS to 1460 does not work with WCCP L2 redirect	3
CSCsh98343	WCCP redirect-list and mask-acl merge results in wrong redirect info	3
CSCsi05906	WCCP:appliance failover does not update TCAM adjacency	3
CSCsi10702	There is no simple backup/restore process to copy tdb from Primary CM	3
CSCsi28118	FTP behavior of Central manager appears to be non-RFC compliant	3
CSCsi65531	Exclude CIFS Requests from Connected Cores from CIFS AD	3
CSCsi66278	NT_STATUS_NO_SUCH_FILE is seen when accessing shares from the DFS root	3
CSCsj32479	WAAS EPM state machine does not handle AUTH3 NTLM authentication	3
CSCsj43783	WAFS CIFS works via configured alias but no pie chart is displayed.	3
CSCsj80333	MAC Client V10.4.10 fails wrie to Win2k3 hidden share finder error -36	3
CSCsk01950	Message to disable Inline Interceptiion settings without Inline interfac	3
CSCsk36732	Message RE Cache error: sub hash table is full appears in the log	3
CSCsk51367	WCCP traffic not denied by access list deny statement on WAE	3
CSCsl68531	transparent rserver stops taking connections	3
CSCsj16259	Explorer copy of a large file, client freeze for tens of seconds	4

WAAS ACE DDTS Previously Encountered Not Fixed

Table L-14 lists WAAS ACE DDTS Previously Encountered Not Fixed by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-14 DDTS Previously Encountered Not Fixed in DCAP 4.0 WAAS ACE Testing

DDTS	Descriptoin	Severity
CSCsi69388	WAE-502: Routing incorrect: Static routes sent out Gig1/0 after reload	2
CSCsi75538	WAE-502: Startup and running config different after copy run start	5

WAAS WCCP DDTS Encountered

Table L-15 lists WAAS WCCP DDTS Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-15 DDTS Encountered in DCAP 4.0 WAAS WCCP Testing

DDTS	Descriptoin	Severity
CSCsi69388	WAE-502: Routing incorrect: Static routes sent out Gig1/0 after reload;	2
CSCsi75538	WAE-502: Startup and running config different after copy run start;	5

WAAS WCCP DDTS of Interest but Not Encountered

Table L-16 lists WAAS WCCP DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-16 DDTS of Interest but Not Encountered in DCAP 4.0 WAAS WCCP Testing

DDTS	Descriptoin	Severity
CSCsg11506	EPM breaks connections if it doesn't intercept both traffic directions	2
CSCsi44131	Wrong WAFS core selected with CIFS AD	2
CSCsk34237	Egress multicast replication broken due to wccp .	2
CSCsf02614	WAFS misses keepalive (PE error code 246)	3
CSCsh93105	Changing MSS to 1460 does not work with WCCP L2 redirect	3
CSCsh98343	WCCP redirect-list and mask-acl merge results in wrong redirect info	3
CSCsi05906	WCCP:appliance failover does not update TCAM adjacency	3
CSCsi10702	There is no simple backup/restore process to copy tdb from Primary CM	3
CSCsi28118	FTP behavior of Central manager appears to be non-RFC compliant	3
CSCsi65531	Exclude CIFS Requests from Connected Cores from CIFS AD	3
CSCsi66278	NT_STATUS_NO_SUCH_FILE is seen when accessing shares from the DFS root	3
CSCsj32479	WAAS EPM state machine does not handle AUTH3 NTLM authentication	3
CSCsj43783	WAFS CIFS works via configured alias but no pie chart is displayed.	3
CSCsj80333	MAC Client V10.4.10 fails wrie to Win2k3 hidden share finder error -36	3
CSCsk01950	Message to disable Inline Interceptiion settings without Inline interfac	3
CSCsk36732	Message RE Cache error: sub hash table is full appears in the log	3
CSCsk51367	WCCP traffic not denied by access list deny statement on WAE	3
CSCsl68531	transparent rserver stops taking connections	3
CSCsj16259	Explorer copy of a large file, client freeze for tens of seconds	4

WAAS WCCP DDTS Previously Encountered Not Fixed

Table L-17 lists WAAS WCCP DDTS Previously Encountered Not Fixed by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-17 DDTS Previously Encountered Not Fixed in DCAP 4.0 WAAS WCCP Testing

DDTS	Descriptoin	Severity
CSCsi69388	WAE-502: Routing incorrect: Static routes sent out Gig1/0 after reload	2
CSCsi75538	WAE-502: Startup and running config different after copy run start	5

Volume 6: Global Site Selector (GSS)

The Global Site Selector (GSS) leverages DNS's distributed services in order to provide high availability to existing data center deployments by incorporating features above and beyond today's DNS services.

The GSS devices are integrated into the existing DCAP topology along with BIND Name Servers and tested using various DNS rules configured on the GSS. Throughout the testing, the GSS receives DNS queries sourced from client machines as well as via DNS proxies (D-Proxies). The Name Server zone files on the D-Proxies are configured to nsfoward DNS queries to the GSS in order to obtain authoritative responses. Time-To-Live (TTL) values associated with the various DNS resource records are observed and taken into consideration throughout the testing.

The tests in this chapter focus on the fundamental ability of the GSS working together with existing BIND Name Servers in order to provide global server load balancing.

The following DDTS types were logged:

- GSS DDTS Filed, page L-33
- GSS DDTS of Interest but Not Encountered, page L-33

GSS DDTS Filed

Table L-18 lists GSS DDTS Filed by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-18 DDTS Filed in DCAP 4.0 GSS Testing

DDTS	Descriptoin	Severity
CSCsk51868	Secondary GSS fails to answer DNS queries;	3

GSS DDTS of Interest but Not Encountered

Table L-19 lists GSS DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-19 DDTS of Interest but Not Encountered in DCAP 4.0 GSS Testing

DDTS	Descriptoin	Severity
CSCsd85917	Show command is broken for keepalive of type kal-ap by tag	2
CSCsd85917	Show command is broken for keepalive of type kal-ap by tag	2
CSCsc46380	From GUI create ans type vip - save & refresh (F5)- java null ptr excptn	3
CSCsc46380	From GUI create ans type vip - save & refresh (F5)- java null ptr excptn	3
CSCsc49495	Not able to suspend the Answer using CLI	3
CSCsc49495	Not able to suspend the Answer using CLI	3
CSCsc49585	Warning message for KAL-AP keepalive is missing - related to VIP address	3
CSCsc49585	Warning message for KAL-AP keepalive is missing - related to VIP address	3
CSCsd93659	GSS not forwarding request when proximity enable	3
CSCsd93659	GSS not forwarding request when proximity enable	3
CSCse78532	TCP traffic NS forwarded by GSS to internal DNS server becomes UDP	3
CSCse78532	TCP traffic NS forwarded by GSS to internal DNS server becomes UDP	3
CSCsd27224	ntp-server <A.B.C.D> is setting system time while NTP is disabled	4
CSCsd27224	ntp-server <A.B.C.D> is setting system time while NTP is disabled	4
CSCsg25654	no password for enable command	enhancement (6)
CSCsg25654	no password for enable command	enhancement (6)

Volume 7: Bladeswitching

The HP c-Class BladeSystem is a complete infrastructure of servers, network management and storage integrated in a modular design, built to deliver the services vital to a business data center. By consolidating these services into a single enclosure, power, cooling, physical space, management, server provisioning and connectivity savings can all be benefited.

In the DCAP topology both the Intel-based BL460c and AMD-based BL465c were provisioned to run the front end Oracle 11i E-Business Suite web application. BL685c servers were provisioned to provide back-end database service with Oracle Real Application Clusters (RAC). VMware ESX 3.0.2 was installed on BL485c servers, which were set up with boot from SAN and clustered to provide VMotioning capabilities. Each ESX server hosted Oracle Web application, Exchange Server 2003 hosts, and Windows Server 2003 domain controllers. The integrated Cisco 3020 Layer 2+ switch provided network connectivity to the data center Aggregation Layer in Data Center A. Four switches were housed in the DCA blade chassis and each one was configured with a dual-port Etherchannel dual homed to the Aggregation Layer switches. The Blade Enclosure in Data Center B was deployed with pass-thru modules allowing each server to connect directly into the Access Layer Catalyst 4948 and 6500 switches. The tests in this chapter focus on the basic feature functionality of the 3020 switch and its response to negative events.

The following DDTS types were logged:

- HP 3020 DDTS Filed, page L-35

HP 3020 DDTS Filed

Table L-20 lists HP 3020 DDTS Filed by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-20 DDTS Filed in DCAP 4.0 Blade Server Testing

DDTS	Descriptoin	Severity
CSCsk62010	Crash at show interfaces vlan 1 switchport;	2
CSCsk65749	Show Platform: BIT-4-OUTOFRANGE:bit 1 is not in the expected range;	3
CSCsk65666	SU07: Show Platform: SUPERVISOR-3-FATAL: MIC exception error 80;	enhancement (6)

Volume 10: Applications: TIBCO Rendezvous

TIBCO Rendezvous (RV) is a multicast-based messaging middleware of particular interest to those financial customers with trading floors as part of their business. TIBCO RV takes financial data feeds in and sends them out to interested receivers subscribed to various multicast groups. The tests in this chapter, performed against TIBCO RV v7.5, verify the functionality of the networking infrastructure in its ability to deliver these messages as well as validating the ability of the network infrastructure to deliver inter-DC multicast data.

The following DDTS types were logged:

- TIBCO Rendezvous DDTS of Interest but Not Encountered, page L-35

TIBCO Rendezvous DDTS of Interest but Not Encountered

Table L-21 lists TIBCO Rendezvous DDTS of Interest but Not Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-21 DDTS of Interest but Not Encountered in DCAP 4.0 TIBCO Rendezvous Testing

DDTS	Descriptoin	Severity
CSCsk34237	Egress multicast replication broken due to wccp .	2

Volume 11: Data Center High Availability

Cisco DCAP 4.0 testing included disaster recovery testing for the Oracle 11i E-Business Suite, Oracle 10gR2 database, and Microsoft Exchange 2003 application test beds described above. The data center disaster recovery tests included failing both applications over to DCb, and then failing the applications back to DCa. Replication of SAN data over fibre channel (with write acceleration enabled) and replication of NAS data over IP (with WAAS optimization) were key enablers.

Failover testing started with a simulation of a disaster by severing all WAN and SAN links to and from DCa. Failback testing started with a controlled shutdown of applications in DCb. Application data created or modified in DCb during failover was replicated back to DCa as part of the failback procedure. Parts of the failover and failback procedures were automated with GSS, ACE, and CSM, and other parts

were manual. For each test, a timeline of automatic and manual steps was constructed and two key metrics, the Recovery Point Objective (RPO) and Recovery Time Objective (RTO), were determined and reported.

The following DDTS types were logged:

- High Availability DDTS Encountered, page L-36

High Availability DDTS Encountered

Table L-22 lists High Availability DDTS Encountered by the DCAP test engineering team during Cisco Data Center Assurance Program (DCAP) 4.0 testing.

Table L-22 DDTS Encountered in DCAP 4.0 High Availability Testing

DDTS	Descriptoin	Severity
CSCsg79439	DRE chunk aggregation performance problem;	3
CSCsh72271	Slow file transfers for files that were repeatedly transfered over time;	3
CSCsl68531	transparent rserver stops taking connections;	3