Cisco Data Center Assurance Program (DCAP) 3.0

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 DCAP team does not exist as a stand-alone 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 3.0 was undertaken by members of the Safe Harbor and NSITE test teams.

Table 1 lists ESE Data Center Design Guides referenced for this release. Where possible and sensible, these design guides are leveraged for various technologies that are 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 3.0

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 File	http://www.cisco.com/application/pdf/en/us/guest/netsol/ns107/c133/ccmigration_09186a0080733855.pdf
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

Table 1 Relevant ESE Design Guides for DCAP 3.0 (continued)

Design Guide	External URL
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/s304/c649/ccmigration_09186a00807ed7e1.pdf

There are other sources of design guidance as well that were leveraged in designing the DCAP 3.0 test environment, including white papers and implementation guides from third-party vendors. For a more robust list of resources used in DCAP 3.0, 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/en/US/products/hw/contnetw/networking_solutions_products_generic_content0900aecd806121d3.html
- http://www.cisco.com/en/US/products/hw/contnetw/networking_solutions_products_generic_content0900aecd806121d3.html
- http://www.cisco.com/go/datacenter
- http://www.cisco.com/en/US/netsol/ns741/networking_solutions_products_generic_content0900aecd8062a61e.html
- (Cisco Internal) http://wwwin.cisco.com/marketing/datacenter/programs/dcap.shtml
- (Cisco Internal) http://safeharbor.cisco.com/

About This Book

Though all of the elements in the data center function as a whole, these elements can also be viewed individually. DCAP 3.0 testing was performed both on the individual technologies and on the data center as a whole. This book consists of 10 chapters and an appendix. Each chapter will focus on a particular component of the data center, with the final chapter focusing on the data center as a whole. The appendix will be used to document procedures and methods used in support of the testing, that may or may not be directly related to the testing itself.

Chapter 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 Network Appliance, Hewlett-Packard and EMC. A summary of software used in this phase of testing is provided here.

Chapter 2: LAN (Layer 2-3) Infrastructure

The DCAP 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. The Catalyst 4948 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.

Chapter 3: LAN (Layer 4-7) Services

The modular Catalyst 6500 switching platform supports various line cards which provide services at Layers 4-7. Several of these Service Modules are bundled together and tested in the DCAP topology, including the Content Switching Module (CSM), Firewall Services Module (FWSM) and Secure Socket Layer Module (SSLM). The tests in this chapter focus on the ability of these three Service Modules to work together to provide load-balancing, security and encryption services to data center traffic.

There were two physically different deployments tested in DCAP 3.0. In one, the Aggregation Layer switches are performing double duty, housing Service Modules and providing aggregation for the Access Layer. In the other, the Service Modules are deployed in separate Service Switches that are connected to the Aggregation Layer switches.

Note Many of the tests reported in this chapter were run twice, once with SSLM version 2.1(11) and once with SSLM version 3.1(1). While previous phases of DCAP had only SSLM version 3.1(1), 2.1(11) was added in this phase to provide coverage for a defect that had been fixed in this version. 3.1(1) does not have the fix for this defect and only 2.1(11) will be tested in the next phase of DCAP.

Chapter 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 multiprotocol SAN director running SAN-OS version 3.1(2).

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.

The topology also includes an ADIC i500 Scalar tape library with two IBM LTO3 tape drives.

Chapter 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 Wide-area Application Engines (WAE) at the both the remote branch and data center WAN edges. The tests in this chapter focus on the basic functionality of the WAAS software on the WAE devices as well as the data center and branch routers ability to intercept and redirect TCP-based traffic.

Note Safe Harbor testing on WAAS 4.0(9)b10 (used in DCAP 3.0) failed Safe Harbor product testing. While 4.0(9)b10 functioned well as part of the DCAP solution, 4.0(11)b24 is recommended for customer deployments. While no Safe Harbor product testing was performed on WAAS 4.0(11)b24, many of the DCAP WAAS tests were re-executed against this newer code (please see Appendix for results).

Chapter 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 GSSes 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.

Chapter 7: Bladeservers

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 and configured to run the Oracle 11i E-Business Suite. The integrated Cisco 3020 Layer 2+ switch provided network connectivity to the data center aggregation layer. The tests in this chapter focus on the basic feature functionality of the 3020 switch and its response to negative events.

Chapter 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) 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 the Database Tier was Active in only one data center with data being replicated synchronously to the second data center making it Active/Passive. The architecture deployed showcases various Cisco products GSS, CSM, MDS which made up the entire solution. Cisco WAAS technologies were leveraged to optimize Oracle application traffic sent from branch offices.

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 included verifying the configuration and functionality of E-Business application integration with GSS, CSM, Active/Active hybrid mode and WAAS optimizations. Tests also covered the failover and failback of the E-Business application in a data center disaster recovery situation.

Chapter 9: Applications: Microsoft Exchange 2003

DCAP 3.0 testing includes Microsoft Exchange 2003. The 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 Network Appliance. This storage was replicated synchronously from the primary to the standby cluster. Tests included running Microsoft LoadSim and 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.

Chapter 10: Data Center Disaster Recovery and Business Continuance

DCAP 3.0 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 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.

CHAPTER 1

Overview

The Safe Harbor team is a key partner for the DCAP team. The methodology and approach to testing that Safe Harbor uses ensures that the testing is relevant and the software is more stable. That is why this methodology has been adopted by the DCAP team for use in its testing.

DCAP Testing Methodology

There are several elements of the Safe Harbor methodology that provide for a higher level of reliability in software releases. First is the deference that Safe Harbor gives to Cisco's customers. The results of every test are viewed from the perspective of how they might impact the end-user. The same goes for the bug scrubs that the Safe Harbor team conducts on a given release candidate. Bugs are monitored prior to a release and during the entire testing cycle. Any defects that may impact a customer are evaluated and scrutinized. Severity 3 defects are given the same level of consideration as Severity 1 and 2 defects, as they might be just as impacting to a customer.

A fix for a given defect always has the potential of causing problems in the same area of code, or even a different area. Because of this possibility of "collateral damage," Safe Harbor will never begin a final run of testing until the last fix has been committed. Only FCS code makes it into the test bed for the final test run. Because the software candidate is already available to the customer, the Safe Harbor team can maintain a Time-to-Quality focus, rather responding to time-to-market pressures.

Lastly, and perhaps most importantly, the Safe Harbor team anchors its testing philosophy with an unqualified openness. Safe Harbor reports the results, as they occurred, so that customers have the opportunity to evaluate them based on their requirements. That is why DCAP aligns itself so closely with this successful Safe Harbor approach.

DCAP Testing Overview

This document presents the results of Cisco DCAP 3.0 testing.

Cisco DCAP 3.0 testing passed. See the DDTS summary tables per chapter for more details on the defects that were encountered or noted during testing.

DCAP 3.0 testing builds on the previous phase by incorporating more data center elements, including:

- Bladeserver testing
- Oracle 11i E-Business Suite
- Microsoft Exchange 2003
- Data center failover testing

This phase of DCAP testing builds on the previous phase by tying many of the individual data center elements more closely together through the use of business applications. While the previous phases of testing focused mainly on the individual performances of siloed technologies such as LAN, SAN, global site load balancing and WAN optimization, DCAP 3.0 delivers an actual end-to-end data center deployment. The addition of two applications was a key deliverable for this phase of testing. Oracle 11i E-business Suite and Microsoft Exchange 2003 were built into the topology to demonstrate how each of these individual elements could work together to provide a robust datacenter deployment. DCAP 3.0 also brought the addition of bladeservers to provide a more real-world environment.

Figure 1-1 gives a very high-level view of the DCAP 3.0 test topology components. Each of the two data centers is similar in the components. Each has a LAN infrastructure consisting of Core, Aggregation, and Access Layers. Servers form the bridge between the LAN and the SAN components, being both LAN-attached (via Ethernet) and SAN-attached (via FibreChannel). The servers are dual-homed into redundant SAN fabrics and the redundant SAN fabrics are, in turn, connected to the storage arrays. The storage layers in both data centers are connected for replication purposes. There are three branch offices as part of the DCAP test topology to provide for remote users.

Figure 1-1 Cisco DCAP 3.0 Test Topology Components

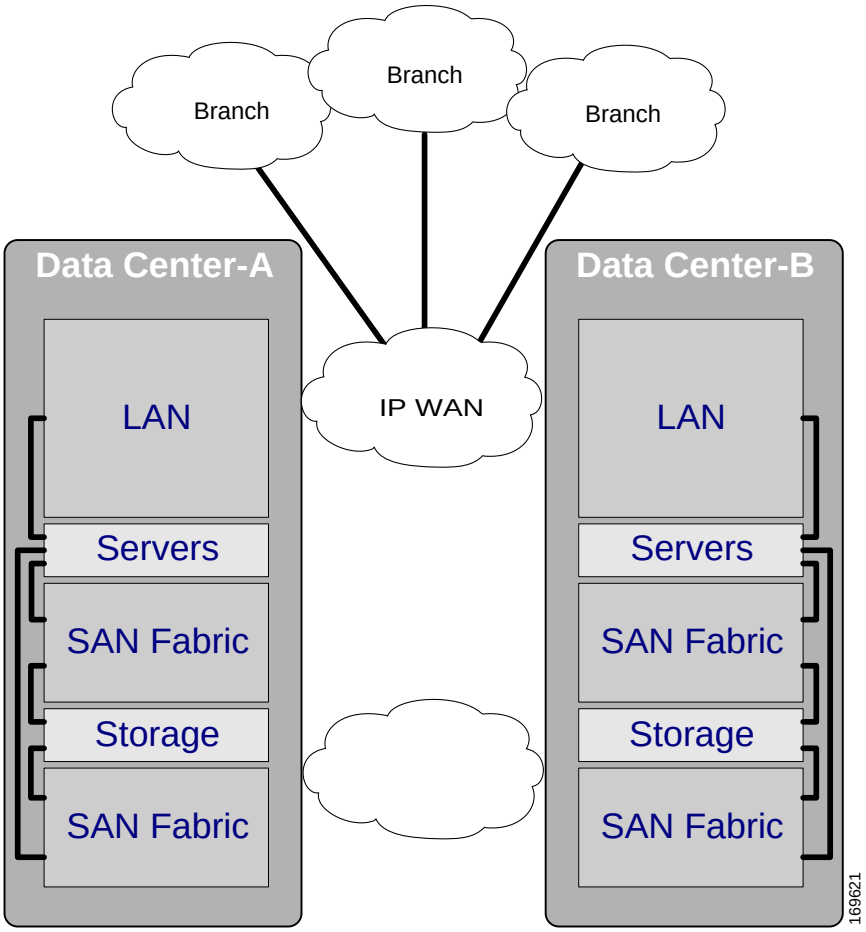
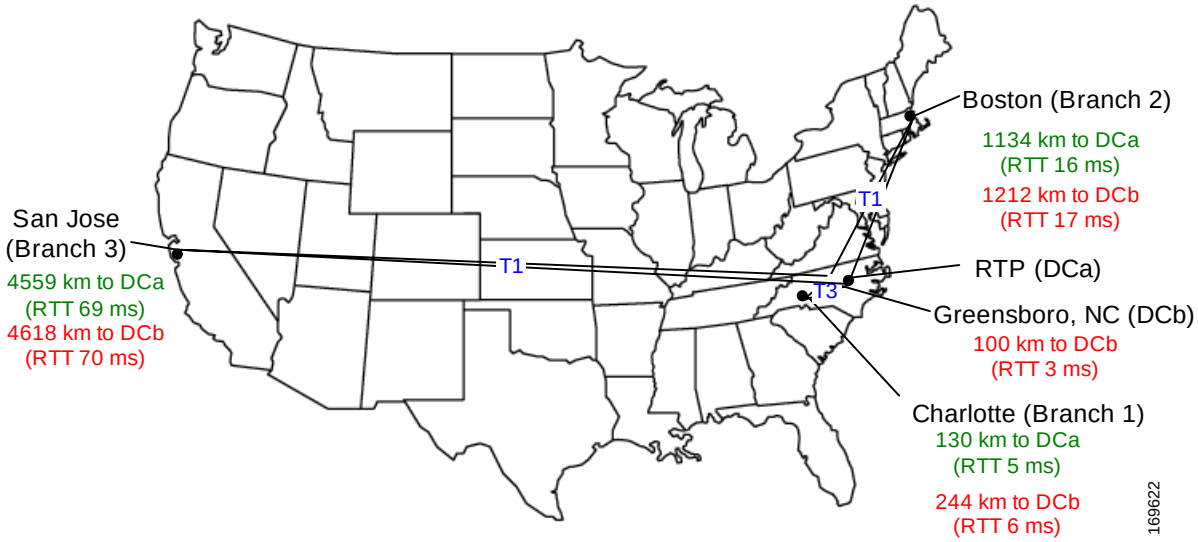


Figure 1-2 demonstrates how the geographic components are laid out, using Research Triangle Park, NC, USA (the location of the main DCAP test lab) as a reference point. In this context, the primary data center is located in RTP, NC and the secondary data center is located in Greensboro, NC, about 100km away from the primary. The three branch offices are located in Charlotte, NC, Boston, MA, and San Jose, CA. The diagram shows the distance and RTT (round trip time) latency between the sites.

Figure 1-2 DCAP Data Center and Branch Map



Note For more information on this multi-site setup, please see the Appendix.

Where possible, DCAP testing tries to stay away from emulation, in favor of real hardware. This is where our relationships with certain vendors becomes key. The DCAP team has worked closely with several vendor partners to provide industry-standard hardware coverage in the DCAP test topology. Table 1-1 shows the vendor hardware that is being used in the DCAP environment.

Table 1-1 Vendor Hardware in DCAP 3.0

Vendor	Hardware	Primary Function in DCAP
Network Appliance	FAS6070 *	File (NAS) and block storage
Hewlett-Packard	XP10000 **	Block storage
Hewlett-Packard	BladeSystem c7000 (c-Class) **	Application servers
EMC	Symmetrix DMX-3 ***	Block storage

* For more information, please visit http://www.netapp.com

** For more information, please visit http://www.hp.com

*** For more information, please visit http://www.emc.com

The DCAP testing effort often relies on testing performed by other teams, particularly the Safe Harbor team. As mentioned above, 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. Table 1-2 lists the various software levels for the various products covered in this phase of DCAP testing. Where possible, EDCS (Cisco internal) document numbers are provided so that the reader can locate and review the results of relevant Safe Harbor product testing. For Cisco customers, please ask your account team for a customer-facing version of these results documents. A comprehensive list of the test cases executed in these other projects is provided in the Appendix to this document.

Table 1-2 Cisco Product Software Used in DCAP 3.0

Platform	Software Version	EDCS Doc. No.
Catalyst 6500: Supervisor 720	12.2(18)SXF7	583951
Firewall Services Module (FWSM)	2.3(3.2)	523606
Content Switching Module (CSM)	4.2(6)	605556
Secure Socket Layer Module (SSLM)	2.1(11) 3.1(1)	566635 * 504167
Catalyst 4948-10GE	12.2(31)SGA	N/A**
Cisco 3020 Gig Switch Module (integrated in HP BladeServers)	12.2(35)SE	N/A **
Cisco WAAS: WAE-502	4.0(9)b10 4.0(11)b24	610852 ***
Cisco WAAS: WAE-512	4.0(9)b10 4.0(11)b24	610852 ***
Cisco WAAS: WAE-612	4.0(9)b10 4.0(11)b24	610852 ***
Cisco WAAS: WAE-7326	4.0(9)b10 4.0(11)b24	610852 ***
Global Site Selector (GSS)	1.3(3)	N/A **
Cisco MDS 9500	3.1(2)	NA **

* The results for SSLM 2.1(10) testing were used, along with undocumented testing on 2.1(11) to cover those areas potentially impacted by a single defect fix in 2.1(11).

** Safe Harbor does not perform regular testing on these platforms.

*** Safe Harbor testing on WAAS 4.0(9)b10 Failed Safe Harbor product testing; While 4.0(9)b10 functioned well as part of the DCAP solution, 4.0(11)b24 is recommended for customer deployments; While no Safe Harbor product testing was performed on WAAS 4.0(11)b24, many of the DCAP WAAS tests were re-executed against this newer code (please see Appendix for results).

Note This documentation stipulates that the tests either Pass, Pass with Exception, or Fail. If a test Fails, and the impact to our customer base is determined to be broad enough, the entire release fails (resulting from 1 or more unresolved defects, notwithstanding unresolved cosmetic, minor, or test-specific defects, which are scrutinized by the DCAP engineering team as being a non-show stopping defect. If a test Fails, and the impact to our customer base is determined to be minor, the release as a whole may still Pass, with defects noted. Exceptions to any particular test are noted for disclosure purposes and incidental noteworthy clarification. Customers are advised to carefully tests review selected, by test suite and feature, particular to their environment.

DCAP Latencies and Bandwidths

The DCAP 3.0 test bed models two data centers and three branches with relative distances and IP WAN round trip times (RTTs) as depicted in the map in Table 1-2.

The RTT for the IP WAN connections was computed as follows:

1. Compute the one-way propagation delay: add 0.5 msec per 100 km (based on the approximate speed of light through fiber).
2. Compute the one-way queuing and switching delay: add approximately 1 msec per 10 msec of propagation delay.
3. Compute the RTT: double the sum of the results from steps 1 and 2.

For example, the RTT between the data centers is 2 times the sum of 0.5 msec and 1 msec or 3 msec.

Table 1-3 summarizes the IP WAN latencies based on averages from the ping command and the bandwidth of the WAN links.

Table 1-3 IP WAN Latencies and Bandwidths

	DCa	DCb	Bandwidth
DCa	-	3 msec	1 Gbps
DCb	3 msec	-	1 Gbps
Branch 1	5 msec	6 msec	T3 (45 Mbps)
Branch 2	16 msec	17 msec	T1 (1.5 Mbps)
Branch 3	69 msec	70 msec	T1 (1.5 Mbps)

The IP WAN latencies and bandwidth were simulated by routing all connections through a RedHat Enterprise Linux 4 update 4 server with five Gigabit Ethernet interfaces and the iproute package installed. The iproute package provides the /sbin/tc (traffic control) command, which enables enforcement of various queueing disciplines on the Ethernet interfaces. One discipline known as netem (for network emulation) allows imposing a delay on traffic transiting the interface. Another discipline called tbf (for token bucket filter) allows restriction of bandwidth. Both of these disciplines were used in the DCAP Phase 3 test bed for emulating a representative IP WAN.

The SAN extension latency between data centers was set at 1 ms, since the queueing and switching delays are negligible. An Anue Systems latency generator with model HSDG192-B OC-192/STM-64 blades was used to simulate the latency.

CHAPTER 2

Layer 2-3 Infrastructure

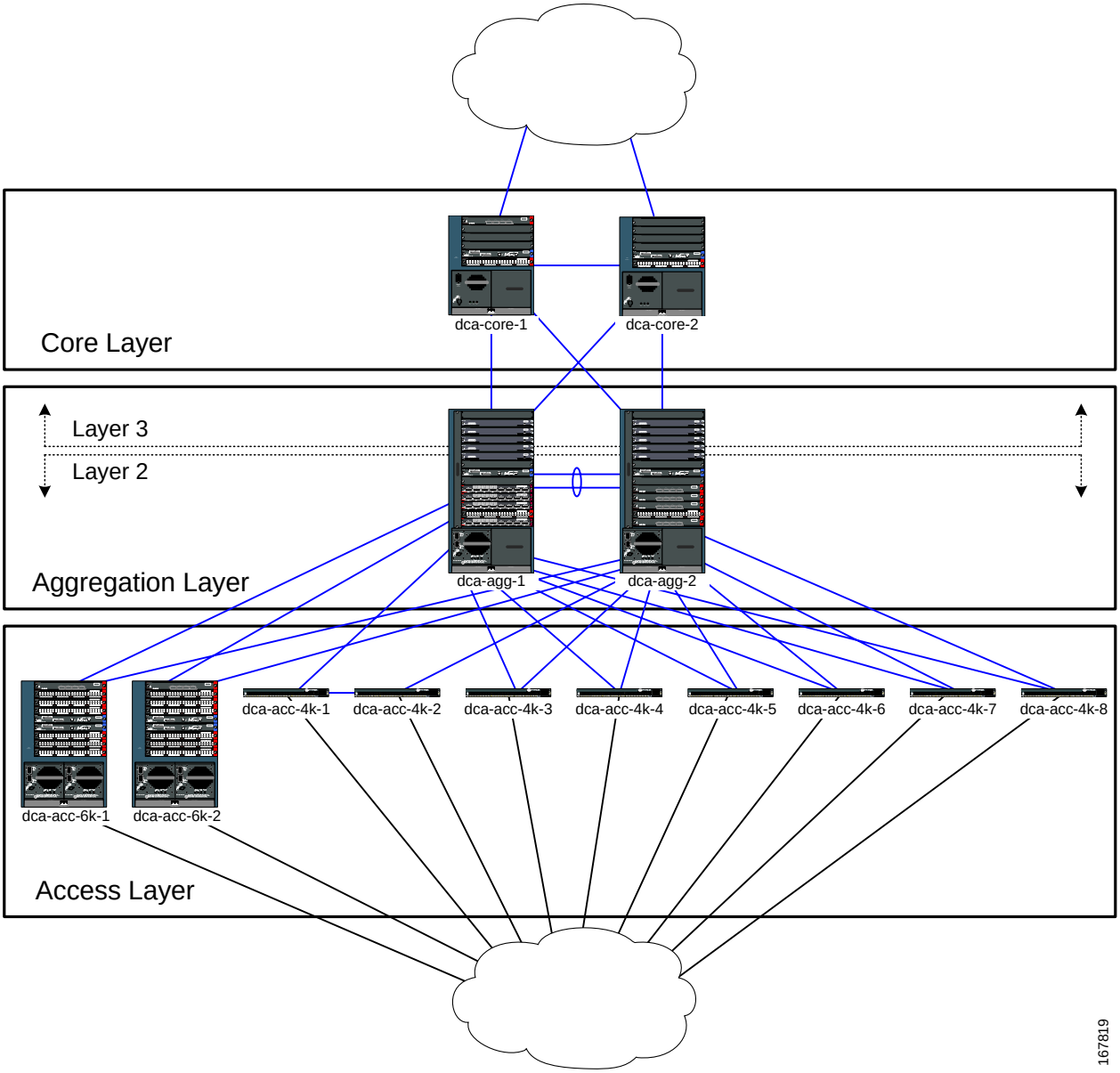
The Cisco DCAP 3.0 topology consists of two separate data centers, DCa and DCb. Each data center has its own LAN, SAN and storage components. Tests performed regarding Layer 2-3 Infrastructure verification were executed against the LAN topology in DCa. Figure 2-1 shows this portion of the test topology. It is divided into three distinct, logical layers called the Core, Aggregation, and Access Layers offering the Layer 2-3 services listed in Table 2-1.

Table 2-1 Cisco DCAP 3.0 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

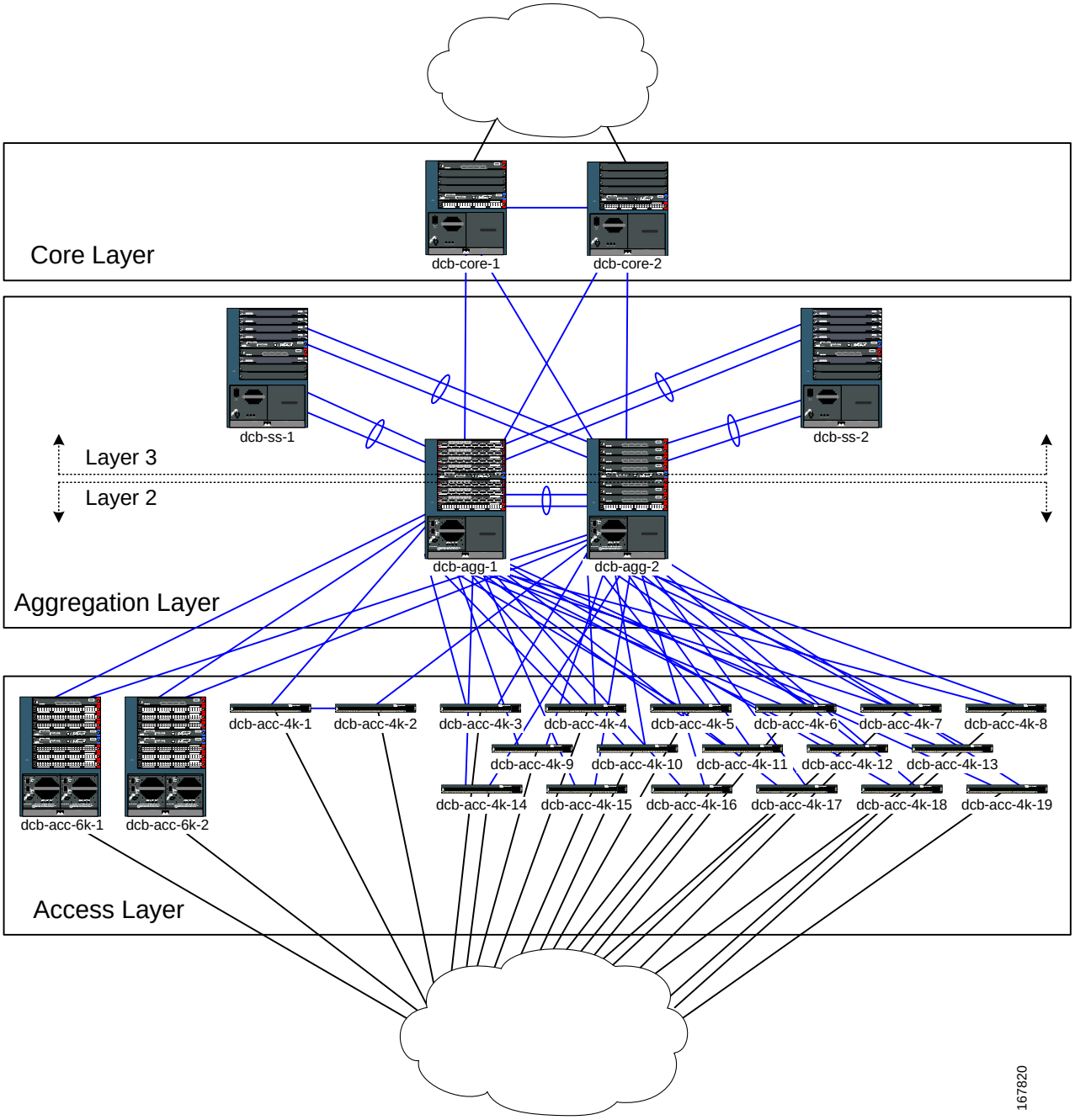
Figure 2-1 shows the Cisco DCAP 3.0 DCa topology.

Figure 2-1 Cisco DCAP 3.0 DCa Topology



The LAN topology in DCb (Figure 2-2) is built differently and more to scale. As will be discussed in a later chapter, the DCb LAN topology is built to accommodate a “Service Switch” 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 switchport 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 19 Catalyst 4948 Access Layer switches, the bulk of which are present to provide for a scaled spanning-tree environment.

Figure 2-2 Cisco DCAP 3.0 DCb Topology



Layer 2 Topology Overview

Figure 2-1 also shows the demarcation between Layer 2 and Layer 3 in the DCa LAN test topology. There are six 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. There are also 6 additional Catalyst 4948 switches present in the topology to provide for 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 six 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, provide 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 (the 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 CSM and FWSM 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 separate modules on each device provides an additional level of redundancy.

Layer 3 Topology Overview

Referring again to Figure 2-1, 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 2-3 Test Results Summary

Table 2-2 summarizes tests executed as part of the Cisco DCAP 3.0 testing initiative. Table 2-2 includes the feature or function tested, the section that describes the feature set the feature or function belongs to, the component tests for each feature or function, and whether the test is new in this phase of DCAP testing.

A number of resources were referenced during the design and testing phases of the L2-3 infrastructure in DCAP. These include the Data Center Infrastructure Design Guide 2.1 and supporting documents, produced by Cisco's Enterprise Solution Engineering Data Center team. Links to these document are directly below. In Table 2-2, where applicable, pointers to relevant portions of these document are provided for reference purposes.

Data Center Infrastructure Design Guide 2.1 (SRND):

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

Note Test results are unique to technologies covered and actual scenarios in which they were tested. DCAP is designed to cover critical path areas and augment ongoing regression and systems testing.

Table 2-2 Cisco DCAP 3.0 L2-3 Testing Summary

Test Suites	Features/Functions	Tests	Results
Baseline, page 2-9	Topology Baseline, page 2-10	1. Topology Baseline	
	Device Management, page 2-11	1. Upgrade of Supervisor 720 System in Core Layer 2. Upgrade of Supervisor 720 System in Aggregation Layer 3. Upgrade of Supervisor 720 System in Access Layer 4. Upgrade of Catalyst 4948-10GE System in Access Layer 5. Upgrade of Content Switching Module (CSM) 6. Upgrade of Firewall Services Module (FWSM) 7. Upgrade of Secure Socket Layer Services Module (SSLSM) 8. General On-Line Diagnostics (GOLD) 9. SNMP MIB Tree Walk 10. Local SPAN 11. Remote SPAN (rSPAN)	
	Device Access, page 2-23	1. Repeated Logins Using SSH Version 1 2. Repeated Logins Using SSH Version 2	
	CLI Functionality, page 2-25	1. CLI Parser Functionality Using SSHv1 2. CLI Parser Functionality Using SSHv2 3. CLI Parser Functionality Using SSHv1 on 4948 4. CLI Parser Functionality Using SSHv2 on 4948	CSCsc81109 CSCsc81109
	Security, page 2-27	1. Malformed SNMP Polling 2. Malformed SSH Packets 3. NMAP Open Port Scan	
	Traffic Forwarding, page 2-30 SRND: Page 2-8	1. Zero Packet Loss 2. Distributed FIB Consistency	
Layer 2 Protocols	Link Aggregation Control Protocol (LACP), page 2-33	1. LACP Basic Functionality 2. LACP Load Balancing	
	Trunking, page 2-35	1. 802.1q Trunking Basic Functionality	
	Spanning Tree, page 2-36 SRND: Page 2-11 SRND: Page 5-1	1. Root Guard	
	Unidirectional Link Detection (UDLD), page 2-40	1. UDLD Detection on 10GE Links	
Layer 3 Protocols	Hot Standby Router Protocol (HSRP), page 2-41 SRND: Page 2-11	1. HSRP Basic Functionality	

Table 2-2 Cisco DCAP 3.0 L2-3 Testing Summary (continued)

Test Suites	Features/Functions	Tests	Results
	Open Shortest Path First (OSPF), page 2-43	1. OSPF Route Summarization 2. OSPF Database Verification	
	IP Multicast, page 2-45	1. Multi-DC Auto-RP with MSDP	

Table 2-2 Cisco DCAP 3.0 L2-3 Testing Summary (continued)

Test Suites	Features/Functions	Tests	Results
Negative Testing	Hardware Failure, page 2-48 SRND: Page 2-11 SRND: Page 6-9 SRND: Page 6-14 SRND: Page 7-6	1. Access Layer Supervisor Failover Using SSO with NSF	
		2. Standby Supervisor Access Layer Repeated Reset	CSCsj67108
		3. Reset of Aggregation Layer Device dca-agg-1	
		4. Reset of Aggregation Layer Device dca-agg-2	
		5. Reset of Core Layer Device dca-core-1	
		6. Reset of Core Layer Device dca-core-2	
		7. Spanning Tree Primary Root Failure & Recovery	
		8. HSRP Failover with Fast Timers	
		9. HSRP Recovery From System Failure	
		10. Failure of EtherChannel Module on dca-agg-1	CSCek26222
		11. Failure of EtherChannel Module on dca-agg-2	CSCek26222
	Link Failure, page 2-65 SRND: Page 2-11 SRND: Page 6-9	1. Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2	
		2. Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1	
		3. Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2	
		4. Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2	
		5. Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1	
		6. Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2	
		7. Failure 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1	
		8. Failure 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2	
		9. Failure 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2	
		10. Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1	
		11. Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2	
		12. Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1	
		13. Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2	
		14. Network Resiliency Test	

Layer 2-3 DDTS Summary

Table 2-3 lists Development Defect Tracking System (DDTS) software bugs with descriptions, and comments filed by the DCAP testing team during Cisco DCAP 3.0 L2-3 Infrastructure testing. Table 2-4 lists DDTS with descriptions encountered during Cisco DCAP 3.0 L2-3 Infrastructure testing.

Table 2-3 Summary of DDTS Filed During Cisco DCAP 3.0 L2-3 Testing

DDTS	Description
CSCsj67108	memory leak in rf task during reset of standby supervisor. Nearly ever time standby Sup720 is reset, lose ~70KB SP memory

Table 2-4 Summary of DDTS Encountered During Cisco DCAP 3.0 L2-3 Testing

DDTS	Description
CSCed73359	PM-4-BAD_COOKIE detected when OIR performed on SSL mod w/o shutdown
CSCek26222	mem leak at tm_dbg_msg_queue_init
CSCsc81109	Trceback when issuing CLI command
FN-62488	WS-C4948-10GE May Reset Due to Multibit ECC Error

Layer 2-3 Infrastructure Test Cases

Functionality critical to global enterprises in Cisco DCAP 3.0 Layer 2-3 testing is described in the following sections. Refer to Cisco DCAP 3.0 Cisco Data Center Assurance Program (DCAP) 3.0 Configurations document for test device configurations.

- Baseline, page 2-9
- Layer 2 Protocols, page 2-32
- Layer 3 Protocols, page 2-41
- Negative Testing, page 2-48

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.

The following test features were conducted:

- Topology Baseline, page 2-10
- Device Management, page 2-11
- Device Access, page 2-23
- CLI Functionality, page 2-25
- Security, page 2-27
- Traffic Forwarding, page 2-30

Topology 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 by comparison . These tests help to establish a baseline level of expected behavior so that real problems can be more easily identified.

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

The following test was performed:

- Topology Baseline, page 2-10

Topology Baseline

This test verified 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 Topology 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 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

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

Results

Topology Baseline passed.

Device Management

Device Management tests 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.

These tests verified that the Cisco IOS upgrade process worked correctly.

The following tests were performed:

- Upgrade of Supervisor 720 System in Core Layer, page 2-12
- Upgrade of Supervisor 720 System in Aggregation Layer, page 2-13
- Upgrade of Supervisor 720 System in Access Layer, page 2-13
- Upgrade of Catalyst 4948-10GE System in Access Layer, page 2-14

Upgrade of Supervisor 720 System in Core 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. Core layer device dca-core-1 was upgraded from 12.2(18)SXF3 Native IOS to 12.2(18)SXF4 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 test follows:

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

- We expect the Cisco IOS Upgrade of Supervisor 720 System in Core Layer upgrade process to complete without error.
- We expect no CPU or memory problems.

Results

Upgrade of Supervisor 720 System in Core Layer passed.

Upgrade of Supervisor 720 System in Aggregation 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. Aggregation layer device dca-agg-1 was upgraded from 12.2(18)SXF3 Native IOS to 12.2(18)SXF4 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 test follows:

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

- We expect the Cisco IOS Upgrade of Supervisor 720 System in Aggregation Layer upgrade process to complete without error.
- We expect no CPU or memory problems.

Results

Upgrade of Supervisor 720 System in Core Layer 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)SXF3 Native IOS to 12.2(18)SXF4 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

- We expect the Cisco IOS Upgrade of Supervisor 720 System in Access Layer upgrade process to complete without error.
- We expect no CPU or memory problems.

Results

Upgrade of Supervisor 720 System in Access Layer passed.

Upgrade of Catalyst 4948-10GE System in Access Layer

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(25)SG Native IOS to 12.2(31)SXG 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 test follows:

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

- We expect the Cisco IOS Upgrade of Catalyst 4948-10GE System in Access Layer upgrade process to complete without error.
- We expect no CPU or memory problems.

Results

Upgrade of Catalyst 4948-10GE System in Access Layer passed.

Upgrade of Content Switching Module (CSM)

This test verified the ability for code to be upgraded for the latest version of Safe Harbor certified CSM code to the version of CSM code that is under test. The CSM in device dca-agg-1 was upgraded from 4.1(5) to 4.1(6) to ensure that all configurations were upgraded without issue.

Test Procedure

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 CSM in dca-agg-1 is running the old CSM image using the **show module 2** command.

Step 4 Verify that the CSM 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 CSM directly from the Supervisor.

Step 6 Set up a session between the supervisor engine and the CSM using the **session slot 2 processor 0** command.

Step 7 Load the image from the supervisor engine to the CSM using the **upgrade** command.

Step 8 Once the new image has completed the download, **exit** the session and reboot the CSM module from the supervisor CLI, using the **hw-module module 2 reset** command.

Step 9 When the CSM module comes back online, use the **show module 2** command on dca-agg-1 to verify that the new image has been loaded.

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

- We expect the Cisco IOS Upgrade of Content Switching Module (CSM) upgrade process to complete without error.
- We expect no CPU or memory problems.

Results

Upgrade of Content Switching Module (CSM) passed.

Upgrade of 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.

Note The FWSM code that is under test in DCAP Phase One is 2.3(3.2). This is the first FWSM code to be Safe Harbor certified. For this reason, in this test, there will be no actual upgrade. Instead, a reload will be performed. The entire procedure for the upgrade remains below, and where steps were skipped, it was noted.

Test Procedure

The procedure used to perform the Upgrade of 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.

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

- We expect the Cisco IOS Upgrade of Firewall Services Module (FWSM) upgrade process to complete without error.
- We expect no CPU or memory problems.

Results

Upgrade of Firewall Services Module (FWSM) passed.

Upgrade of Secure Socket Layer Services Module (SSLSM)

This test verified the ability for code to be upgraded for the latest version of Safe Harbor certified SSLSM code to the version of SSLSM code that is under test. The SSLSM in device dca-agg-1 is already running the current version, hence SSLSM is upgraded from 3.1(1) to 3.1(1) to ensure that upgrade works without issue with the current supervisor version.

Test Procedure

The procedure used to perform the Upgrade of Secure Socket Layer Services Module (SSLSM) test follows:

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 SSLSM in dca-agg-1 is running the old SSLSM image using the **show module 3** command.

Step 4 Verify that the SSLSM 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 SSLSM directly from the Supervisor.

Step 6 Boot the SSLSM blade in slot 3 on dca-agg-1 into the maintenance partition using the **hw-module module 3 reset cf:1** command.

Step 7 Verify that the the slot 3 SSLSM is in the maintenance partition using the **show module 3** command.

The **Sw** field in this command output should be appended with a lower-case **m** and the **Status** field should real **Ok**.

Step 8 Download the new image onto the slot 3 SSLSM using the **copy tftp: pclc#3-fs:** command. Enter **127.0.0.71** when prompted for the TFTP location.

Step 9 Once the download has completed successfully, you will see a message printed to the console of dca-agg-1 reading **You can now reset the module**. Reboot the slot 3 SSLSM using the **hw-module module 3 reset** command.

Step 10 Once the SSLSM has come back online, verify that it is running the new image using the **show module 3** 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

- We expect the Cisco IOS Upgrade of Secure Socket Layer Services Module (SSLSM) upgrade process to complete without error.
- We expect no CPU or memory problems.

Results

Upgrade of Secure Socket Layer Services Module (SSLSM) passed.

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

- We expect that the complete set of online diagnostics will have been run on all modules in the systems under test, as configured.
- We expect no CPU or memory problems.

Results

General On-Line Diagnostics (GOLD) 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

- We expect that no tracebacks or crashes to occur on the DUT.
- We expect no CPU or memory problems.

Results

SNMP MIB Tree Walk passed.

Local SPAN

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 test follows:

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-acc-6k-2, 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-acc-6k-2 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

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

Results

Local SPAN passed.

Remote SPAN (rSPAN)

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) test follows:

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 Te9/4 using the**monitor session 2 source interface Te9/4 both** command. By specifying **both**, the session will SPAN ingress and egress traffic on Te9/4.

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

- 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 sustained or unexpected impact on the CPU or memory.

Results

Remote SPAN (rSPAN) 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.

The following tests were performed:

- Repeated Logins Using SSH Version 1, page 2-23
- Repeated Logins Using SSH Version 2, page 2-24

Repeated Logins Using SSH Version 1

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 test follows:

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

- We expect 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 passed.

Repeated Logins Using SSH Version 2

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 test follows:

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

- We expect 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 passed.

CLI Functionality

Parser testing 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).

The following tests were performed:

- CLI Parser Functionality Using SSHv1, page 2-25
- CLI Parser Functionality Using SSHv2, page 2-25
- CLI Parser Functionality Using SSHv1 on 4948, page 2-26
- CLI Parser Functionality Using SSHv2 on 4948, page 2-27

CLI Parser Functionality Using SSHv1

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 test follows:

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

- 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 passed with exception CSCsc81109.

CLI Parser Functionality Using SSHv2

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 test follows:

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

- 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 passed with exception CSCsc81109.

CLI Parser Functionality Using SSHv1 on 4948

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 test follows:

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

- 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 passed.

CLI Parser Functionality Using SSHv2 on 4948

This test verified **show** and **clear** commands on dca-acc-4k-2 through the use of an automated script. 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

- 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.

Security

Resistance 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.

The following tests were performed:

- Malformed SNMP Polling, page 2-27
- Malformed SSH Packets, page 2-28
- NMAP Open Port Scan, page 2-29

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

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

- We expect all DUTs to not hang, crash, or give tracebacks.
- We expect no CPU or memory problems.

Results

Malformed SNMP Polling passed.

Malformed SSH Packets

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 test follows:

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

- We expect that SSH vulnerability testing does not cause the router to reload, hang, or crash.
- We expect no CPU or memory problems.

Results

Malformed SSH Packets 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

- We expect open ports revealed by the NMAP tool as presumed.
- We expect no CPU or memory problems.

Results

NMAP Open Port Scan passed.

Traffic Forwarding

Traffic forwarding measures basic traffic forwarding features and abilities of the DCAP 1.0 test topology.

The following tests were performed:

- Zero Packet Loss, page 2-30
- Distributed FIB Consistency, page 2-31

Zero Packet Loss

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 test follows:

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

- We expect traffic loss due to background test traffic to be tolerable.
- We expect no CPU or memory problems.

Results

Zero Packet Loss passed.

Distributed FIB Consistency

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 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 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

- We expect total parity between the FIB maintained on the supervisor (PFC) and the FIBs maintained on the DFCs.
- We expect no CPU or memory problems.

Results

LACP Basic Functionality passed.

Layer 2 Protocols

Layer 2 testing looks at several key protocols running at the Data Link Layer (Layer 2) of the OSI model, in the Cisco DCAP 3.0 test topology.

The following test features were conducted:

- Link Aggregation Control Protocol (LACP), page 2-33
- Trunking, page 2-35
- Spanning Tree, page 2-36

- Unidirectional Link Detection (UDLD), page 2-40

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.

The following tests were performed:

- LACP Basic Functionality, page 2-33
- LACP Load Balancing, page 2-34

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 lacp
channel-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

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

Results

LACP Basic Functionality passed.

LACP Load Balancing

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 test follows:

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

- We expect traffic to 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 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.

The following test was performed:

- 802.1q Trunking Basic Functionality, page 2-35

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

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

Results

802.1q Trunking Basic Functionality passed.

Spanning Tree

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.

The following tests were performed:

- Rapid PVST+ Basic Functionality, page 2-36
- Root Guard, page 2-38

Rapid PVST+ Basic Functionality

In the Data Center test topology dca-agg-1 is configured to be the primary root switch for all VLANs, while 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 test follows:

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 is 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

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

Results

Rapid PVST+ Basic Functionality 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

- We expect the interfaces with STP root guard enabled to transition to "Root Inconsistent" state when they begin receiving BPDUs 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 BPDUs.
- We expect no CPU or memory problems.

Results

Root Guard 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.

The following test was performed:

- UDLD Detection on 10GE Links, page 2-40

UDLD Detection on 10GE Links

This test forced a unidirectional link condition on one of the 10-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 10-Gigabit Ethernet links, Te13/1 and Te13/2 (on each device). On dca-agg-1, the tx fibers of Te13/1 and Te13/2 were switched, creating a crossed-fiber situation.

This test verified that 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 10GE 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 Te13/1 and Te13/2 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 Te13/1 and Te13/2 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 Te13/1 and Te13/2 on dca-agg-1 and dca-agg-2.

Step 6 Return the transmit fibers to their original location and flap interface Te13/2 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 Te13/1 and Te13/2 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

- 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 shut down for the port determined to be unidirectional.
- We expect no CPU or memory problems.

Results

UDLD Detection on 10GE Links passed.

Layer 3 Protocols

Layer 3 tests look at several key protocols running at the Network Layer (Layer 3) of the OSI model in the DCAP test topology.

The following test features were conducted:

- Hot Standby Router Protocol (HSRP), page 2-41
- Open Shortest Path First (OSPF), page 2-43

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.

The following test was performed:

- HSRP Basic Functionality, page 2-42

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

- 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.

The following tests were performed:

- OSPF Route Summarization, page 2-43
- OSPF Database Verification, page 2-44

OSPF Route Summarization

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 occurred 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 test follows:

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 101.1.0.0 255.255.0.0 area 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 Te9/1** command to verify that this device has an interface with a /30 address matching the 172.31.1.x format.

The IP address of interface Te9/1 is 172.31.1.6/30.

Step 4 On dca-voodoo-2, an Area Border Router (ABR) in Area 20, use the **show ip route 172.31.1.6** command to verify that the route to this address has been summarized as a /24.

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

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

The IP address of interface VLAN 1101 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 read "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

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

Results

OSPF Route Summarization passed.

OSPF Database Verification

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 test follows:

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

- We expect the OSPF database to populated with the correct number of LSAs of each type for the topology being used.

Results

OSPF Database Verification passed.

IP Multicast

Traditional IP communication allows a host to send packets to a single host (unicast transmission) or to all hosts (broadcast transmission). IP multicast provides a third scheme, allowing a host to send packets to a subset of all hosts (group transmission). These hosts are known as group members. Packets delivered to group members are identified by a single multicast group address. Multicast packets are delivered to a group using best-effort reliability, just like IP unicast packets.

The multicast environment consists of senders and receivers. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message. A multicast address is chosen for the receivers in a multicast group. Senders use that address as the destination address of a datagram to reach all members of the group. Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time.

The following test was performed:

- Multi-DC Auto-RP with MSDP, page 2-46

Multi-DC Auto-RP with MSDP

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 configure to use Auto-RP and MSDP(Multi Source Discovery Protocol). 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-DC Auto-RP with MSDP test follows:

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 the Auto-RP candidate dca-core-1, we see three lines of configuration for the RP information:

a. A static configuration naming the anycast address 10.10.0.1 as the RP for groups defined by access-list 3. ACL 3 defines two groups, 224.0.1.39 and 224.0.1.40, which are Auto-RP control groups. 224.0.1.39 is the group on which candidate RP's announce themselves and mapping agents listen. Once an RP is elected, the group 224.0.1.40 is used for other routers in the network, which join this group at startup (by default), to learn of the elected RP address and the groups for which that is the RP.

b. All Auto-RP's 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(224.x.x.x - 239.x.x.x).

c. 3. dca-core-1 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 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 informations 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.

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 sparse-mode PIM. 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 sparse-mode PIM. 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

- We expect the OSPF database to populated with the correct number of LSAs of each type for the topology being used.

Results

OSPF Database Verification passed.

Negative Testing

Negative tests measure the response of the test topology to various negative events, such as simulated hardware failures, link failures and whole device failures.

The following test features were conducted:

- Hardware Failure, page 2-48
- Link Failure, page 2-65

Hardware Failure

Hardware failure testing measures the ability of the DCAP test topology to absorb various hardware failures, including system crashes and module resets.

The following test was performed:

- Access Layer Supervisor Failover Using SSO with NSF, page 2-49
- Standby Supervisor Access Layer Repeated Reset, page 2-50
- Reset of Aggregation Layer Device dca-agg-1, page 2-51
- Reset of Aggregation Layer Device dca-agg-2, page 2-52
- Reset of Core Layer Device dca-core-1, page 2-53
- Reset of Core Layer Device dca-core-2, page 2-54
- Spanning Tree Primary Root Failure & Recovery, page 2-55
- HSRP Failover with Fast Timers, page 2-58
- HSRP Recovery From System Failure, page 2-61
- Failure of EtherChannel Module on dca-agg-1, page 2-62
- Failure of EtherChannel Module on dca-agg-2, page 2-64

Access Layer Supervisor Failover Using SSO with NSF

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 test follows:

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

- We expect the system to recover from each forced failover successfully.
- We expect the failed supervisor to come back online in the SSO standby role following each failover.
- We expect each failover to 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 passed.

Standby Supervisor Access Layer Repeated Reset

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 Standby Supervisor Access Layer Repeated Reset test follows:

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 dcb-acc-6k-1 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 dcb-acc-6k-1 (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 nine 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 fail overs are complete, analyze the traffic statistics for an excessive number of errors and to verify that none of the fail overs resulted in more than three 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

- We expect 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

Standby Supervisor Access Layer Repeated Reset passed with exception CSCsj67108.

Reset of Aggregation Layer Device dca-agg-1

The Aggregation Layer provides the bulk of services for traffic coming into the datacenter. 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 very fast SSO/NSF failover. This is in-line with the goals of predictability with regards to traffic in the datacenter. If anything less than all services fail over, the result is a very 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 lay in 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 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 Aggregation Layer Device dca-agg-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 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

- We expect traffic loss of five seconds or less for each failover performed.
- We expect traffic forwarding to resume when the reset device comes back online.
- We expect no CPU or memory problems.

Results

Reset of Aggregation Layer Device dca-agg-1 passed.

Reset of Aggregation Layer Device dca-agg-2

The Aggregation Layer provides the bulk of services for traffic coming into the datacenter. Services such as server load balancing, SSL decryption and encryption, and fire walling 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 very fast SSO/NSF failover. This is in-line with the goals of predictability with regards to traffic in the datacenter. If anything less than all services fail over, the result is a very 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 lay in 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 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 Aggregation Layer Device dca-agg-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 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

- We expect traffic loss of three seconds or less for each failover performed.
- We expect failed supervisors to come back online without manual intervention.
- We expect no CPU or memory problems.

Results

Reset of Aggregation Layer Device dca-agg-2 passed.

Reset of Core Layer Device dca-core-1

The Core Layer is the first stop for traffic coming into the datacenter. 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 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-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 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

- We expect raffic loss of three seconds or less for each failover performed.
- We expect traffic forwarding to resume when the reset device comes back online.
- We expect no CPU or memory problems.

Results

Reset of Core Layer Device dca-core-1 passed.

Reset of Core Layer Device dca-core-2

The Core Layer is the first stop for traffic coming into the datacenter. 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 test follows:

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

- We expect raffic loss of three seconds or less for each failover performed.
- We expect traffic forwarding to resume when the reset device comes back online.
- We expect no CPU or memory problems.

Results

Reset of Core Layer Device dca-core-2 passed.

Spanning Tree Primary Root Failure & Recovery

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.

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. There are 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 & 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 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

- 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 no CPU or memory problems.

Results

Spanning Tree Primary Root Failure & Recovery passed.

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 two seconds. The configured Dead Timer for each VLAN is six seconds.

Should dca-agg-1 fail, dca-agg-2 will assume the active HSRP router role after six 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 six 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 six 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 two seconds for the Hellos and six 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 two seconds for the Hellos and six 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 four to six 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 six 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

- We expect Standby HSRP router to become Active following a loss of 2-3 hello PDUs.
- 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 no CPU or memory problems.

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 six 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

- We expect HSRP to failover correctly when the Active HSRP router is taken out of service.
- We expect that, when the router is brought back online, it will resume its Active HSRP role after the configured 60-second delay.
- We expect no CPU or memory problems.

Results

HSRP Recovery From System Failure passed.

Failure of EtherChannel Module on dca-agg-1

The Port-channel between the two Aggregation Layer devices, dca-agg-1 and dca-agg-2, is critical for health monitoring and connection synchronization connections between the service modules. This link, an 802.1q trunk, carries the VLANs that communicate the heartbeat messages between the CSMs and the FWSMs. Further, it replicates the connection states between the peer service modules so that downtime due to failover is minimized. With that in mind, the redundancy of this port-channel is essential.

In the DCAP test topology, this port-channel is composed of two 10-Gigabit Ethernet 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 test follows:

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.

Individual CPU and memory usage graphs for all devices not under test are at the following location:

Expected Results

- We expect the port-channel interface to maintain normal operation as a logical interface when one of the hardware modules is reloaded.
- We expect traffic loss due to the module reload to be minimal.

- We expect no CPU or memory problems.

Results

Failure of EtherChannel Module on dca-agg-1 passed with exception CSCek26222.

Failure of EtherChannel Module on dca-agg-2

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 test follows:

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

- We expect the port-channel interface to maintain normal operation as a logical interface when one of the hardware modules is reloaded.
- We expect traffic loss due to the module reload to be minimal.
- We expect no CPU or memory problems.

Results

Failure of EtherChannel Module on dca-agg-2 passed with exception CSCek26222.

Link Failure

Link failure testing measures the impact of a link failure occurring in the data path. The ability of the data center infrastructure to respond favorably to such scenarios is critical to the high availability of network resources.

The following tests were performed:

Failure of Single Bundled 10-Gigabit Ethernet Link Between dca-agg-1 and dca-agg-2

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 test follows:

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 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

- We expect the port-channel interface to maintain normal operation as a logical interface when a single bundled link is flapped.
- We expect traffic loss due to the link failure to be minimal.
- We expect 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 passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1

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 test follows:

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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-1 passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2

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 test follows:

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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-agg-2 passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2

This test verified the impact on traffic from a failure of the 10-Gigabit Ethernet link between the two Core Layer devices, dca-core-1 and dca-core-2. Web traffic is sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/1 on dca-core-1 is shut down for 2 minutes then brought back online for 2 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-core-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 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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-1 and dca-core-2 passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1

This test verified the impact on traffic from a failure of the 10-Gigabit Ethernet link between an Aggregation Layer device, dca-agg-1, and a Core Layer device, dca-core-2. Web traffic is sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/2 on dca-core-2 is shut down for 2 minutes then brought back online for 2 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-2 and dca-agg-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 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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-1 passed.

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2

This test verified the impact on traffic from a failure of the 10-Gigabit Ethernet link between an Aggregation Layer device, dca-agg-2, and a Core Layer device, dca-core-2. Web traffic is sent using the Shenick test tool from 100 clients to a VIP on the CSM. Te1/3 on dca-core-2 is shut down for 2 minutes then brought back online for 2 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-2 and dca-agg-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 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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure of 10-Gigabit Ethernet Link Between dca-core-2 and dca-agg-2 passed.

Failure 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1

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 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-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 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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-4k-1 passed.

Failure 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2

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 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-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 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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-4k-2 passed.

Failure 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2

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 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-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 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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure 10 Gigabit Ethernet Link Between dca-acc-4k-1 and dca-acc-4k-2 passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1

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 test follows:

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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-1 passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2

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 test follows:

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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-1 and dca-acc-6k-2 passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1

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 test follows:

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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-1 passed.

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2

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 test follows:

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

- We expect traffic loss due to the link failure to be minimal.
- We expect traffic loss due to the link recovery to be minimal.
- We expect no CPU or memory problems.

Results

Failure of 10 Gigabit Ethernet Link Between dca-agg-2 and dca-acc-6k-2 passed.

Network Resiliency Test

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 test follows:

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

- 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 no CPU or memory problems.

Results

Network Resiliency Test passed.

CHAPTER 3

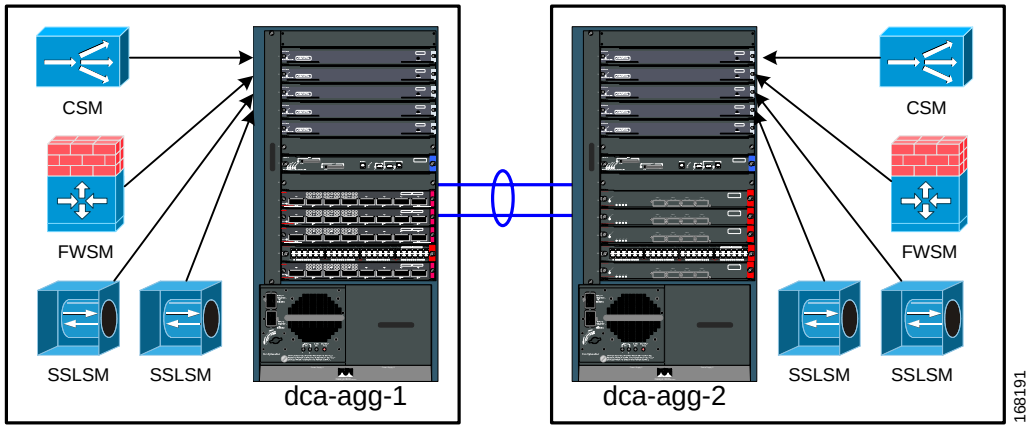
Layer 4-7 Services

Services at Layers 4-7 were handled by Service Modules installed in the Catalyst 6500 platform at the Aggregation Layer. The Content Switching Module (CSM) was installed to provide server load-balancing services. The Firewall Services Module (FWSM) was used to provide basic firewall security services. The Secure Socket Layer Module (SSLM) was used to provide encryption services, both client-facing, and in the backend, as well. The modular design of the Catalyst 6500 platform allows for each of these Service Modules to be installed in the same chassis.

Integrated Bundle Vs. Service Switch Models

In this phase of testing, two models for Service Module deployment are used. In the first one the Service Module bundle is contained directly in the Aggregation Layer switches, dca-agg-1 and dca-agg-2, as can be seen in Figure 3-1. This integrated bundle testing is performed on the DCa LAN topology. In the DCb LAN, a Service Switch model is employed in which an additional pair of Catalyst 6500 switches, outside of dca-agg-1 and dca-agg-2, are deployed to house the Service Modules. In the integrated bundle configuration, the Aggregation Layer devices perform double duty, providing Layer 4-7 services to data center traffic and providing aggregation density to the Access Layer switches. In the Service Switch configuration, these two functions are cleaved.

Figure 3-1 Overview of Integrated SM Bundle in DCa



These switches are both Catalyst 6509s with one slot used for the Supervisor 720 and a second slot used for a 10-Gigabit Ethernet 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. While

only the CSM, FWSM, and SSLM were tested in this phase, many other Service Modules are available, such as the Intrusion Detection Services Module (IDSM), Network Analysis Module (NAM, which was installed, but not tested), and Communications Media Module (CMM), to name a few.

In the integrated bundle model, as configured in the DCa LAN topology, a Catalyst 6513 chassis was used as the Aggregation Layer switch, so that a full 5 slots could be allocated for 10-Gigabit Ethernet linecards. (If a Catalyst 6509 had been used, 5 slots would have been used for the four Service Modules and Supervisor, leaving only 4 slots for 10-Gigabit Ethernet modules.)

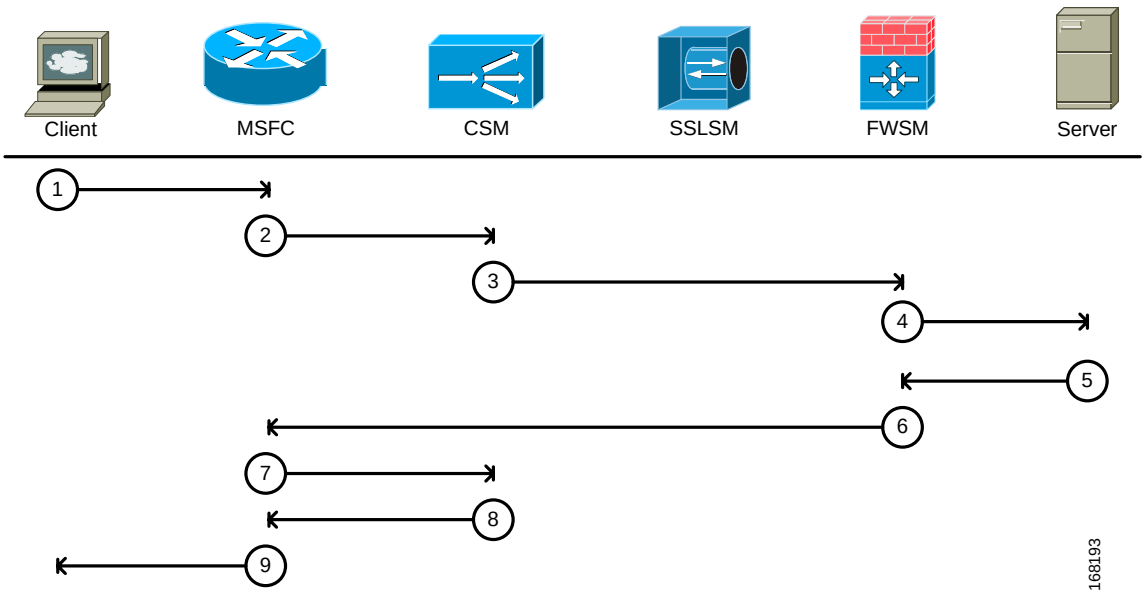
With the Service Modules in their own chassis in the Service Switch deployment, the Aggregation Layer switch is freed up to provide additional 10-Gigabit Ethernet density to the Access Layer. The Catalyst 6513 still only provides 5 slots of 10-Gigabit Ethernet density, though, since only slots 9-13 are dual-fabric capable, a requirement for the 10-Gigabit Ethernet modules. Therefore, a Catalyst 6509 was used as the Aggregation Layer devices, dcb-agg-1 and dcb-agg-2, providing a full 8 slots for 10-Gigabit Ethernet density.

Traffic Pathways Through the Bundle

While later sections of this chapter will discuss the specifics of each Service Module bundle deployment, the way that they handle data center traffic and provide services is the same.

The operation of the CSM for the majority of the traffic it passes is fairly straightforward as seen in Figure 3-2. A client request is sent to the advertised VIP (1) and reaches that destination after being routed by the MSFC (2). The CSM sends the request on to a real server in the serverfarm that is matched with the advertised VIP (3). The request hits the FWSM and is bridged from the outside VLAN to the inside VLAN and forwarded along to the real server (4), which services the request. The server sends the reply to its default gateway on the MSFC (5). After being bridged again by the FWSM (6), this reply hits the policy map configured on the MSFC which tells it to send the reply along to the CSM (7). The CSM replaces the real server's IP address with that of the VIP and sends the reply on to the client (8), again via the MSFC (9).

Figure 3-2 HTTP Request Packet Path Through Service Module Bundle



The DCAP test topology is set up to handle clear text backend and SSL backend. The first case is described above. Figure 3-3 shows an encrypted client request sent to the advertised VIP (1) which reaches that destination after being routed by the MSFC (2). The CSM sends the request on to one of the operational SSLMs in the SSLM serverfarm for decryption (3). The decrypted traffic is sent back to the CSM (4) where it hits another VIP and is thus sent on, in clear text, to the selected real server (5). The request hits the FWSM and is bridged from the outside VLAN to the inside VLAN and forwarded along to the real server (6), which services the request. The server sends the reply to its default gateway on the MSFC (7). After being bridged again by the FWSM (8), this reply hits the policy map configured on the MSFC which tells it to send the reply along to the CSM (9). The CSM works with the SSLM in reverse order, sending the reply first to the SSLM for encryption (10). The SSLM encrypts the reply, using the same client keypair that it used to decrypt it earlier, and sends it back to the CSM (11). The CSM replaces the real server's IP address with that of the VIP and sends the encrypted reply on to the client (12), again via the MSFC (13).

Figure 3-3 HTTPS Request Packet Path Through Service Module Bundle with No Backend Encryption

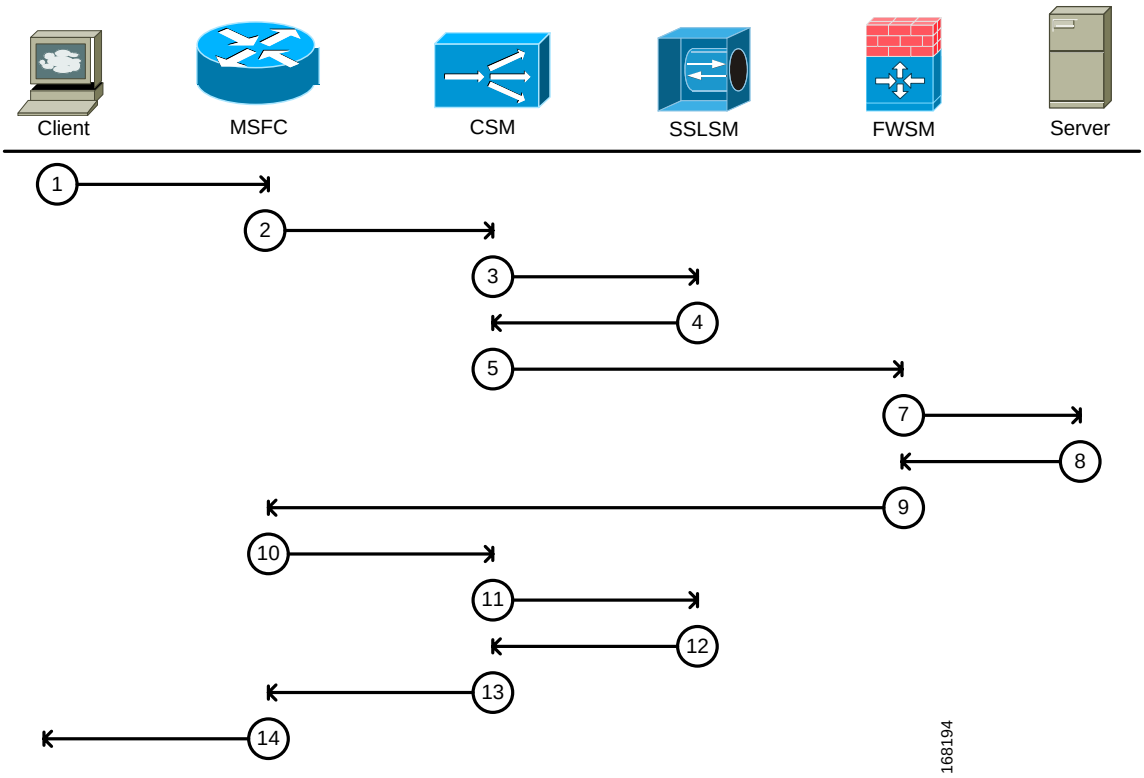
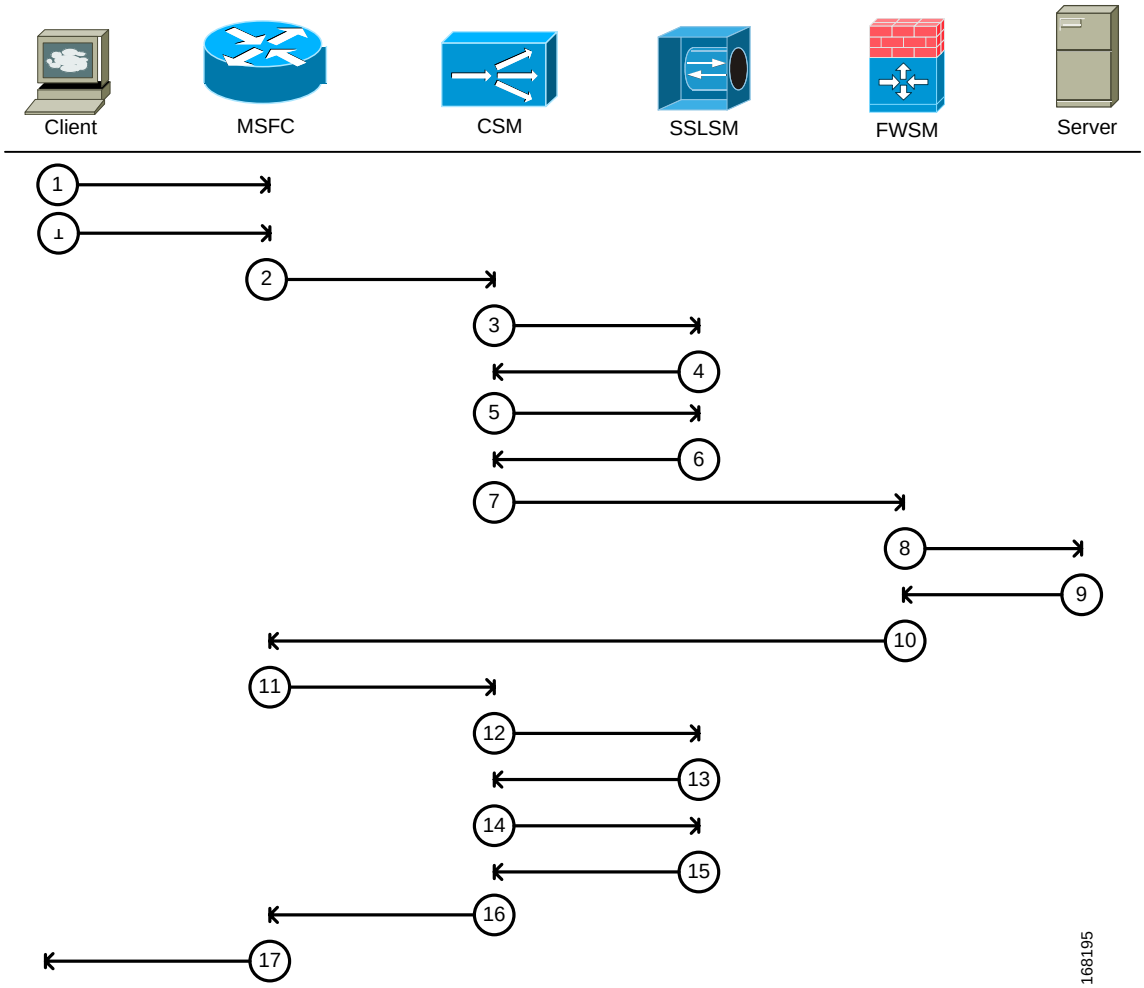


Figure 3-4 show extra steps for traffic that requires backend encryption services. An encrypted client request is sent to the advertised VIP (1) and reaches that destination after being routed by the MSFC (2). The CSM sends the request on to one of the operational SSLMs in the SSLM serverfarm for decryption (3). The decrypted traffic is sent back to the CSM (4) where it hits another VIP that has it sent back to the SSLM for backend encryption (5). The SSLM re-encrypts the request, this time with the server keypair, and sends it back to the CSM (6). The CSM sends the request along to the appropriate real server, encrypted (7). This encrypted request hits the FWSM and is bridged from the outside VLAN to the inside VLAN and forwarded along to the real server (8), which services the request. The server sends the reply to its default gateway on the MSFC (9). After being bridged again by the FWSM (10), this reply hits the policy map configured on the MSFC which tells it to send the reply along to the CSM (11). The

CSM works with the SSLM in reverse order, sending the reply first to the SSLM for decryption (12). The SSLM decrypts the reply using the server keypair and sends the reply, in clear text, back to the CSM (13). The reply hits another VIP on the CSM which sends it back to the SSLM (14), one final time, for re-encryption using the client keypair. Once re-encrypted, the reply is sent back to the CSM (15). The CSM replaces the real server's IP address with that of the VIP and sends the encrypted reply on to the client (16), again via the MSFC (17).

Figure 3-4 HTTPS Request Packet Path Through Service Module Bundle with Backend Encryption

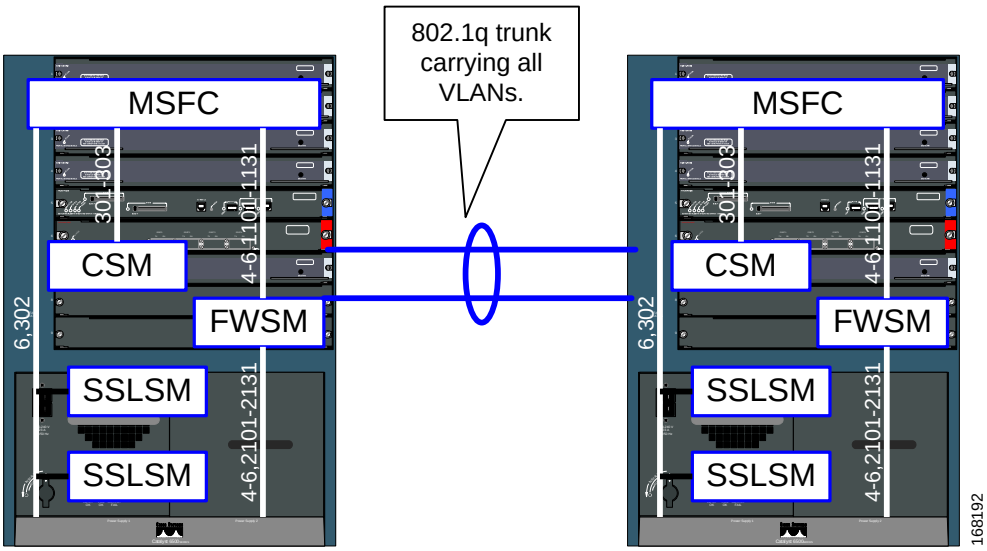


Integrated Bundle Configuration

Though the Service Switch configuration is physically different than the integrated bundle configuration, logically, they provide the Layer 4-7 services to the data center traffic in the same way. The internal workings of the integrated bundle will be explained here first, then the differences in the Service Switch will be outlined.

Each of the aggregation devices in the integrated bundle contains an equal number of Service Modules: one CSM, one FWSM and two SSLMs, as illustrated in Figure 3-5. 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.

Figure 3-5 Logical functionality of services in the integrated bundle configuration



There are several modes that the CSM can be configured to operate in, including bridged and routed modes. In the DCAP test topology, 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 the DCAP DCa LAN 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. As mentioned previously, an active/active condition can wreak havoc on a network when both CSMs begin to advertise the same service. This is why the etherchannel between the two Aggregation Layer devices is so critical.

There are useful statistics listed below regarding the configuration of the CSM services in the DCAP test topology. For redundancy to be operational, each CSM must have exactly the same configuration, with regards to the services that it provides.

- 2000+ real servers are configured and operational

- 30+ server farms group the real servers (30 server farms with 64 reals, 1 with 80)
- 30+ vservers providing one VIP/gateway apiece
- SSLM serverfarms and vservers facilitating the correct handling of HTTPS traffic
- Various flavors of server load-balancing algorithms (round robin, least connections, most connections, weighted server, etc.)
- HTTP, Passive FTP and Active FTP
- TCP probes used in each serverfarm to monitor the health of each of the 2000+ real servers in the Layer 2 domain

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 CSM, the FWSM also uses heartbeat messages to communicate with its peer, verifying its existence. Also like the CSM, only one FWSM can be active at a time, with FWSM software version 2.3(3.2). 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 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.

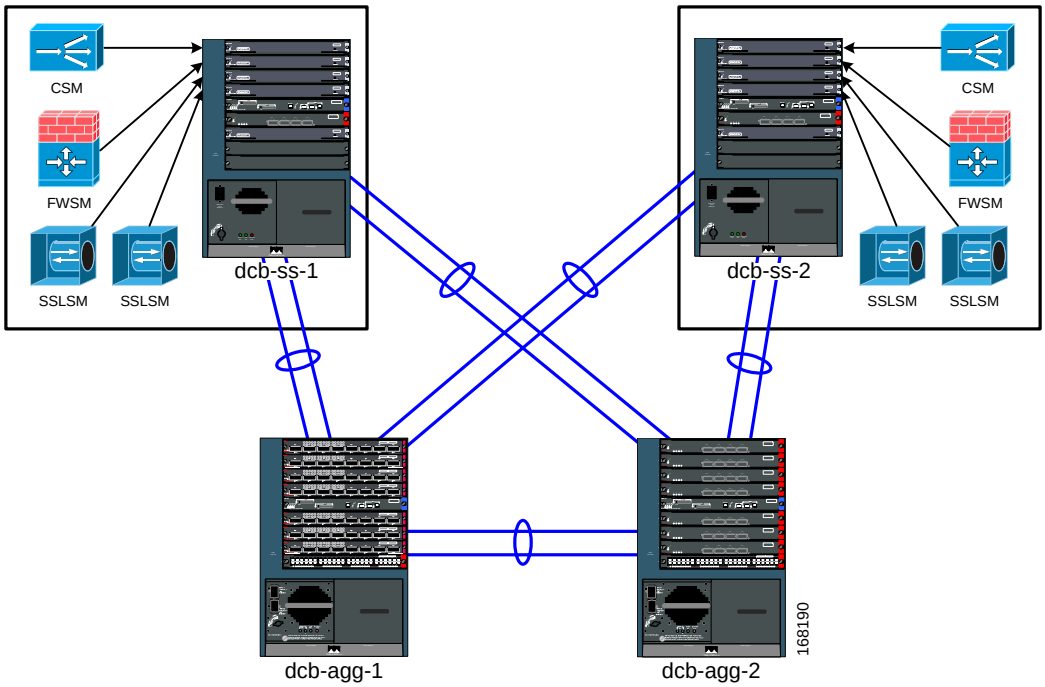
There are four SSLMs in the DCAP DCa LAN test topology, two in each of the Aggregation Layer devices. The SSLMs are neither active nor standby; they work in a pool to handle encryption services for data center traffic. Each of the CSMs is configured with a serverfarm called "SSLSM" which contains four "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.

Service Switch Configuration

Logically, the Service Module bundle in the Service Switch 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 Switch is quite different, as can be seen in Figure 3-6.

Figure 3-6 Overview of Service Switch topology in DCb



There are two Service Switches, dcb-ss-1 and dcb-ss-2. As in the integrated bundle switches dca-agg-1 and dca-agg-2, these Service Switches have a single CSM, a single FWSM, and two SSLMs. Internally, they are connected in the same way. Externally, each Service Switch 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 Switches 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 Switches 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 Switch model. For those inside VLANs whose networks are advertised out to the world, the Service Switches 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 Switch bundle, dcb-agg-1 and dcb-agg-2 share the HSRP responsibilities, as in the integrated bundle setup.)

Layer 4-7 Test Results Summary

Table 3-1 summarizes tests executed as part of the Cisco DCAP 3.0 testing initiative. Table 3-1 includes the feature or function tested, the section that describes the feature set the feature or function belongs to, the component tests for each feature or function, and whether the test is new in this phase of DCAP testing.

In DCAP 3.0 two versions of SSLM code were tested: 2.1(11) and 3.1(1). Version 2.1(11) has the fix for defect CSCsh79045 which can affect environments where header inserts are performed when the HTTP header spans multiple packets. Version 3.1(1) does not have this fix (it is fixed in the 3.1 train in 3.1(2). Several tests were executed multiple times so both SSLM versions were exercised.

In DCAP 4.0 testing, only 2.1(11) code will be used.

A number of resources were referenced during the design and testing phases of the Layer 4-7 Services in DCAP. These include the Server Farm Security in the Business Ready Data Center Architecture v2.1 design guide, produced by Cisco's Enterprise Solution Engineering Data Center team. A link to this document is directly below. In Table 3-1, where applicable, pointers to relevant portions of this document are provided for reference purposes.

Server Farm Security in the Business Ready Data Center Architecture v2.1 (SRND):

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns376/c649/ccmigration_09186a008078e021.pdf

Note Test results are unique to technologies covered and actual scenarios in which they were tested. DCAP is designed to cover critical path areas and augment ongoing regression and systems testing.

Table 3-1 Cisco DCAP 3.0 L4-7 Testing Summary

Test Suites	Features/Functions	Tests	Results
Aggregation Bundle with SSLM 2.1.11	CSM/FWSM Integration, page 3-10 SRND: Page 83 SRND: Page 115 SRND: Page 124	1. Active FTP Through FWSM and CSM 2. Passive FTP Through FWSM and CSM 3. ICMP to a CSM Layer 3 and Layer 4 Vserver 4. DNS Query Through CSM and FWSM 5. FWSM and CSM Layer 4 SYN Attack 6. Idle Timeout UDP	
	CSM/SSLSM Integration, page 3-21 SRND: Page 26-27 SRND: Page 147	1. Backend SSL 2. SSL Sticky 3. URL Rewrite 4. DC UrlRewrite Spanning Packets 5. SSLM CIPHERS 6. DC Cookie Sticky Spanning Packets	

Table 3-1 Cisco DCAP 3.0 L4-7 Testing Summary (continued)

Test Suites	Features/Functions	Tests	Results
Aggregation Bundle with SSLM 2.1.11	Redundancy, page 3-29 SRND: Page 94 SRND: Page 96 SRND: Page 137	1. FWSM Redundancy 2. CSM Redundancy 3. SSLM Reset 4. HSRP Failover	
Aggregation Bundle with SSLM 3.1.1	CSM/SSLSM Integration, page 3-37 SRND: Page 147	1. Backend SSL 2. SSL Sticky 3. URL Rewrite	
	Redundancy, page 3-41 SRND: Page 94 SRND: Page 96 SRND: Page 137	1. FWSM Redundancy 2. CSM Redundancy 3. SSLM Reset 4. HSRP Failover	
Service Switch Bundle with SSLM 2.1.11	CSM/SSLSM Integration, page 3-49 SRND: Page 147	1. Backend SSL 2. SSL Sticky 3. URL Rewrite	
	Redundancy, page 3-54 SRND: Page 94 SRND: Page 96 SRND: Page 137	1. FWSM Redundancy 2. CSM Redundancy 3. SSLM Reset 4. HSRP Failover	
	CSM/FWSM Integration, page 3-63 SRND: Page 119	1. Active FTP Through FWSM and CSM 2. Passive FTP Through FWSM and CSM 3. ICMP to a CSM Layer 3 and Layer 4 Vserver 4. DNS Query Through CSM and FWSM 5. FWSM CSM Layer4 SYN Attack 6. Idle Timeout UDP	
	CSM/SSLSM Integration, page 3-73 SRND: Page 41 SRND: Page 121	1. Backend SSL 2. SSL Sticky 3. URL Rewrite	
	Redundancy, page 3-78 SRND: Page 94 SRND: Page 96 SRND: Page 137	1. FWSM Redundancy 2. CSM Redundancy 3. SSLM Reset 4. HSRP Failover	

Layer 4-7 DDTS Summary

Table 3-2 lists Development Defect Tracking System (DDTS) software bugs with descriptions, and comments filed by the DCAP testing team during Cisco DCAP 3.0 L4-7 testing.

Table 3-2 Summary of DDTS Filed During Cisco DCAP 3.0 L4-7 Testing

CSCsj62985	Minor mem leak in SpanTree Helper RP process when FWSM reset

Layer 4-7 Test Cases

Functionality critical to global enterprises in Cisco DCAP 3.0 L4-7 testing is described in the following sections. Refer to Cisco Data Center Assurance Program (DCAP) 3.0 Configurations document for test device configurations.

SSLM 2.1(11), CSM 4.2(6), FWSM 2.3(3.2)

- Aggregation Bundle with SSLM 2.1.11, page 3-10
- Service Switch Bundle with SSLM 2.1.11, page 3-49

SSLM 3.1(1), CSM 4.2(6), FWSM 2.3(3.2)

- Aggregation Bundle with SSLM 3.1.1, page 3-37
- Service Switch Bundle with SSLM 3.1.1, page 3-63

Aggregation Bundle with SSLM 2.1.11

The following tests verified various functional aspects of the three Service Modules (CSM, FWSM and SSLSM) in a bundled fashion; that is, working together in the same chassis to provide services to data center traffic. Three Service Modules are bundled together in the Aggregation Layer switches in DCa, dca-agg-1 and dca-agg-2.

The following test features were conducted:

SSLM 2.1(11), CSM 4.2(6), FWSM 2.3(3.2)

- CSM/FWSM Integration, page 3-10
- CSM/SSLSM Integration, page 3-21
- Redundancy, page 3-29

CSM/FWSM Integration

CSM/FWSM integration looks at interoperability capacities of the CSM and FWSM, in terms of how they work together to handle data traffic.

The following tests were performed:

- Active FTP Through FWSM and CSM, page 3-63

Active FTP Through FWSM and CSM

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.

Test Procedure

The procedure used to perform the Active FTP Through FWSM and CSM test follows:

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 log
```

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 access-list outgoing 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

- 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 fix up protocol ftp 21 is disabled.
- We expect the CSM vserver to properly load balance Active FTP.

Results

Active FTP Through FWSM and CSM passed.

Passive FTP Through FWSM and CSM

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.

Test Procedure

The procedure used to perform the Passive FTP Through FWSM and CSM test follows:

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.

The output of the command shows that the serverfarm that is being used is FARM1 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.

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 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, 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

- 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 unacceptable impact on the CPU or memory of the DUT.

Results

Passive FTP Through FWSM and CSM passed.

ICMP to a CSM Layer 3 and Layer 4 Vserver

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.

Test Procedure

The procedure used to perform the ICMP to a CSM Layer 3 and Layer 4 Vserver test follows:

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 log
```

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

- We expect the CSM will NOT 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 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. 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 test follows:

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 In the FWSM Vlan1101-2101 context, use the **show fixup** 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 farm1-a 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 STT/TUB 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 FARM1.

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 **show module csm 2 real sfarm farm1-a detail** command to verify that each real server in the serverfarm has made some connections.

Step 12 Issue the **clear module csm 2 counters** and **clear module csm 2 connections** commands on the active CSM to clear connections and counters.

Step 13 Use STT/TUB to send a DNS queries at a rate of roughly 1000/second to vserver VIP-DNS for domain name dcb-penguin2.dcb-dcap.cisco.com.

Step 14 While traffic is running, issue the **show xlate** and **show conn | include most** commands on the Vlan1101-2101 FWSM context to verify the xlate table and number of open connections.

Step 15 Verify the results of the DNS query traffic.

Step 16 Use the **show module csm 2 vserver name vip-dns detail** and **show module csm 2 stats** commands on the active CSM to verify the DNS traffic was properly load balanced.

Counters **Tot matches** and **L4 Load Balanced Decisions** should have roughly the same value. Verify the **Tot matches** counter equals roughly the number of attempts from the test tool.

Step 17 Issue the **clear module csm 2 counters** and **clear module csm 2 connections** commands on the active CSM to clear connections and counters.

Step 18 Use STT/TUB to send a DNS queries at a rate of roughly 1500/second to vserver VIP-DNS for domain name dcb-penguin2.dcb-dcap.cisco.com.

Step 19 While traffic is running, issue the **show xlate** and **show conn | include most** commands on the Vlan1101-2101 FWSM context to verify the xlate table and number of open connections.

Step 20 Verify the results of the DNS query traffic.

Step 21 Use the **show module csm 2 vserver name vip-dns detail** and **show module csm 2 stats** commands on the active CSM to verify the DNS traffic was properly load balanced.

Counters **Tot matches** and **L4 Load Balanced Decisions** should have roughly the same value. Verify the **Tot matches** counter equals roughly the number of attempts from the test tool.

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

- We expect the FWSM to permit DNS traffic to vserver VIP-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

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.

Test Procedure

The procedure used to perform the FWSM and CSM Layer 4 SYN Attack test follows:

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 20
```

You 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

- We expect the FWSM to intercept syn packets being sent to the CSM reals by issuing a syn cookie.
- We except 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 passed.

Idle Timeout UDP

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 Idle Timeout UDP test follows:

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 serverfarm name farm1-a detail** 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.

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

- We expect flows that exceed the idle timeout will be cleared.
- We expect the CSM vserver to properly load balance UDP traffic.

Results

Idle Timeout UDP passed.

CSM/SSLSM Integration

CSM/SSLSM integration looks at interoperability capacities of the CSM and SSLSM, in terms of how they work together to handle data traffic.

The following tests were performed:

Backend SSL

This test verified that the CSM and SSLM successfully work together to load balance SSL traffic on the client side internally decrypted the traffic for advanced Layer 7 processing then re encrypt the traffic load balancing to the backend servers. This test also verified the CSM is able to stick clients to the same real based on cookies.

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 Backend SSL test follows:

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 SSLM's to clear statistics:

```
clear ssl-proxy stats service
clear ssl-proxy stats hdr
clear ssl-proxy stats ssl
```

Step 3 Use the following commands on all four SSLM's to verify statistics have been cleared:

```
show ssl-proxy stats service
show ssl-proxy stats hdr
show ssl-proxy stats ssl client
```

Step 4 Use 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 Use the following commands to verify the counters have been cleared:

```
show mod csm 2 vserver name VIP-HOSTS-SSL detail
show mod csm 2 vserver name VIP-HOSTS-PSSL detail
show mod csm 2 real sfarm FARM1-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.

Step 9 Send multiple HTTPS get requests for 1.gif, 2.gif, and 3.gif from the outside client to vserver VIP-HOSTS-SSLBE. The client emulation tool will generate the traffic using three different cookies.

Step 10 Wait until client emulation traffic has completed, then issue the **show mod csm 2 vservers name VIP-HOSTS-SSLBE detail** command to verify the Tot matches counter equals 600.

Step 11 Issue the **show mod csm 2 vservers name VIP-HOSTS-BE detail** command to verify the Tot matches counter has incremented for the following three policies:

```
100 times for 1.GIF
200 times for 2.GIF
300 times for (default)
```

Step 12 Use the **show mod csm 2 real sfarm FARM1-BE detail** command on the active CSM to verify the load balancing of connections.

Step 13 Use 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 Use the **show ssl-proxy stats service BACKEND30** command on all SSLM's to verify the following two counters equal 600:

```
conns attempted
conns completed
```

Step 16 Issue the following commands on sslm-1 to very conns attempted and conns completed counter have incremented and there are no errors.

```
show ssl-proxy stats service BACKEND30
show ssl-proxy stats service SSL-backend
```

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

- We expect the CSM to correctly load balance SSL traffic.
- We expect the CSM to apply the correct L7 policy on clear text traffic.
- We expect the CSM to be able to stick based on the client cookie.
- We expect the SSLSM to re-encrypt 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

Backend SSL passed.

SSL Sticky

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 SSL Sticky test follows:

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 VIP-HOSTS-SSLBE detail
show mod csm 2 real sfarm FARM1-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 SSLM's to verify ssl-proxy service is operational.

Step 6 Issue the **clear ssl-proxy stats service** command on all four SSLM's to clear ssl-proxy service statistics.

Step 7 Issue the **show ssl-proxy stats service** command on all four SSLM's 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 30 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** command on one of the servers should be some value greater than one 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 SSLM's in serverfarm SSLMSM:

```
show mod csm 2 vservers name VIP-HOSTS-SSLBE detail
show mod csm 2 real sfarm FARM1-BE detail
show mod csm 2 real sfarm SSLSM 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

- We expect the CSM to stick clients to the same real based on SSL Session ID.

Results

Backend SSL passed.

URL Rewrite

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.

Test Procedure

The procedure used to perform the URL Rewrite test follows:

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 DMZ1-WEB with the url-rewrite policy DMZ1-WEB by issuing the **policy url-rewrite dmz1-web** 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 url-rewrite** command on both SSL Modules.

Step 4 From the outside client use the client emulator to generate an HTTPS request to vserver DMZ1-HTTPS. 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 url-rewrite
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 url-rewrite
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

- We expect that the SSLM can rewrite the server issued 300 series redirects from HTTP to HTTPS.

Results

URL Rewrite passed.

DC UrlRewrite Spanning Packets

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. The HTTP requests are sent using a very small MTU (Maximum Transmission Unit), which caused these requests to span many packets.

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.

Test Procedure

The procedure used to perform the DC UrlRewrite 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 150 bytes in order to simulate a dial-up or satellite user.

Step 3 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 4 Clear SSL-proxy service statistics and URL statistics on each of the four SSLSM's in the topology by using the following commands:

```
clear ssl-proxy stats service url-rewrite context Default
clear ssl-proxy stats url
```

Step 5 Verify the SSL-proxy service statistics and URL statistics have been cleared on each of the four SSLSM's in the topology by using the following commands:

```
show ssl-proxy stats service url-rewrite context Default
show ssl-proxy stats url
```

Step 6 Issue the **clear mod csm 2 count** command on the active CSM to clear CSM counters.

Step 7 From the outside client use the client emulator to generate 1000 HTTPS request to vserver SSL-REWRITE.

Step 8 When client emulated traffic has completed, issue the **show ssl-proxy stats url** command on all SSLM's to verify the Rewrites Succeeded counter has incremented to a combined total of 1000.

Step 9 Issue the **show ssl-proxy stats service url-rewrite** command on all SSLM's to verify the conns attempted and full handshakes counters have incremented to 1000.

Step 10 On the active CSM verify the total matches counter for vserver SSL-REWRITE and vserver CLEAR-REWRITE equals 2000 by using the **show mod csm 2 vserver name name detail** command.

Step 11 On the active CSM verify traffic was evenly load balanced between all reals in serverfarm SSLSM and serverfarm FARM30 by using the **show mod csm 2 real sfarm***name***detail** 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

- We expect the SSLM is able to rewrite the server issued 300 series redirects from HTTP to HTTPS. The http requests were sent using a very small MTU (Maximum Transmission Unit), which caused these requests to span many packets.

Results

DC UrlRewrite Spanning Packets passed.

SSLM CIPHERS

This test verified that the SSLM terminated SSL sessions using the correct cipher specified in the ssl proxy. Different ciphers were listed in the ssl proxy in order to verify the client browser negotiated the asymmetric and symmetric ciphers correctly without error.

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 dcap-frontend
```

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 dcap-forward
show ssl-proxy stats service dcap-forward 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 "dcap-frontend" are set to **rsa-with-rc4-128-md5** and that **version all** is selected.

Step 8 On Brnach3-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 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 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 dcap-frontend detail"** command.

Step 12 Modify the ssl ciphers that are configured on the SSLM for ssl-proxy service "dcap-frontend" 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 **branch3-client-1** and make sure it is listening on TCP port 443.

Step 14 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 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 dcap-frontend detail"** command.

Step 17 Modify the ssl ciphers that are configured on the SSLM for ssl-proxy service "dcap-frontend" 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 dcap-forward
clear ssl-proxy session service dcap-frontend
clear ssl-proxy conn service dcap-frontend
```

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 dcap-frontend detail"** command.

Step 21 Modify the ssl ciphers that are configured on the SSLM for ssl-proxy service "dcap-frontend" 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 dcap-forward
clear ssl-proxy session service dcap-frontend
clear ssl-proxy conn service dcap-frontend
```

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://dcap-frontend**

Step 24 Verify the symmetric operations and cipher suites chosen by the SSLM by issuing the **"show ssl-proxy stats service dcap-frontend 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

- We expect the SSLM to terminate SSL connections using various ciphers.
- We expect the SSLM to terminate SSL traffic without error.
- We expect the SSLM to select the correct cipher based on the configured ciphers in the ssl proxy service.

Results

SSLM CIPHERS 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.

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 reqeusts 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

- 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.

Redundancy

The Catalyst 6500 series of switches provide excellent reliability and network stability by offering a number of Hardware Redundancy options. Dual Supervisor Modules were tested by command or physical (OIR) failure testing.

The following tests were performed:

- FWSM Redundancy, page 3-29
- CSM Redundancy, page 3-31
- SSLM Reset, page 3-34
- HSRP Failover, page 3-36

FWSM Redundancy

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 test follows:

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-HTTP 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-HTTP 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-HTTP 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-HTTP 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

- We expect the FWSM to replicate flow information from active to standby FWSM.
- We expect the standby FWSM will transition to active state with the failure of the active FWSM.

Results

FWSM Redundancy passed.

CSM Redundancy

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 follows:

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-HTTP 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-HTTP 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-HTTP 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-HTTP 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-HTTP 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-HTTP 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

- 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 passed.

SSLM Reset

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.

Test Procedure

The procedure used to perform the SSLM Reset test follows:

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 all
```

Issue 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 SSL30.

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

- 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 passed.

HSRP Failover

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 test follows:

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

- We expect the mean failover time for HSRP to be less then the default dead time of 10 seconds.
- We expect that when the failed system becomes operational again, it will resume HSRP active status and forward traffic.

Results

HSRP Failover passed.

Aggregation Bundle with SSLM 3.1.1

The following tests verified various functional aspects of the three Service Modules (CSM, FWSM and SSLSM) in a bundled fashion; that is, working together in the same chassis to provide services to data center traffic. Three Service Modules are bundled together in the Aggregation Layer switches in DCa, dca-agg-1 and dca-agg-2.

The following test features were conducted:

SSLM 3.1(1) CSM 4.2(6), FWSM 2.3(3.2)

- CSM/SSLSM Integration, page 3-37
- Redundancy, page 3-41

CSM/SSLSM Integration

CSM/SSLSM integration looks at interoperability capacities of the CSM and SSLSM, in terms of how they work together to handle data traffic.

The following tests were performed:

- Backend SSL, page 3-73
- SSL Sticky, page 3-75
- URL Rewrite, page 3-76

Backend SSL

This test verified that the CSM and SSLM successfully work together to load balance SSL traffic on the client side internally decrypted the traffic for advanced Layer 7 processing then re encrypt the traffic load balancing to the backend servers. This test also verified the CSM is able to stick clients to the same real based on cookies.

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 Backend SSL test follows:

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 SSLM's to clear statistics:

```
clear ssl-proxy stats service context Default
clear ssl-proxy stats hdr
clear ssl-proxy stats ssl
```

Step 3 Use the following commands on all four SSLM's to verify statistics have been cleared:

```
show ssl-proxy stats service
show ssl-proxy stats hdr
show ssl-proxy stats ssl client
```

Step 4 Use 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 Use the following commands to verify the counters have been cleared:

```
show mod csm 2 vserver name VIP-HOSTS-SSL detail
show mod csm 2 vserver name VIP-HOSTS-PSSL detail
show mod csm 2 real sfarm FARM1-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.

Step 9 Send multiple HTTPS get requests for 1.gif, 2.gif, and 3.gif from the outside client to vserver VIP-HOSTS-SSLBE. The client emulation tool will generate the traffic using three different cookies.

Step 10 Wait until client emulation traffic has completed, then issue the **show mod csm 2 vservers name VIP-HOSTS-SSLBE detail** command to verify the Tot matches counter equals 600.

Step 11 Issue the **show mod csm 2 vservers name VIP-HOSTS-BE detail** command to verify the Tot matches counter has incremented for the following three policies:

```
100 times for 1.GIF
200 times for 2.GIF
300 times for (default)
```

Step 12 Use the **show mod csm 2 real sfarm FARM1-BE detail** command on the active CSM to verify the load balancing of connections.

Step 13 Use 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 Use the **show ssl-proxy stats service BACKEND30** command on all SSLM's to verify the following two counters equal 600:

```
conns attempted
conns completed
```

Step 16 Issue the following commands on sslm-1 to very conns attempted and conns completed counter have incremented and there are no errors.

```
show ssl-proxy stats service BACKEND30
show ssl-proxy stats service SSL-backend
```

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

- We expect the CSM to correctly load balance SSL traffic.
- We expect the CSM to apply the correct L7 policy on clear text traffic.
- We expect the CSM to be able to stick based on the client cookie.
- We expect the SSLSM to re-encrypt 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

Backend SSL passed.

SSL Sticky

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 SSL Sticky test follows:

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 VIP-HOSTS-SSLBE detail
show mod csm 2 real sfarm FARM1-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 SSLM's to verify ssl-proxy service is operational.

Step 6 Issue the **clear ssl-proxy stats service** command on all four SSLM's to clear ssl-proxy service statistics.

Step 7 Issue the **show ssl-proxy stats service** command on all four SSLM's 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 30 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** command on one of the servers should be some value greater than one 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 SSLM's in serverfarm SSLMSM:

```
show mod csm 2 vservers name VIP-HOSTS-SSLBE detail
show mod csm 2 real sfarm FARM1-BE detail
show mod csm 2 real sfarm SSLSM 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

- We expect the CSM to stick clients to the same real based on SSL Session ID.

Results

SSL Sticky passed.

URL Rewrite

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.

Test Procedure

The procedure used to perform the URL Rewrite test follows:

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 url-rewrite** 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

- We expect that the SSLM can rewrite the server issued 300 series redirects from HTTP to HTTPS.

Results

URL Rewrite passed.

Redundancy

The resiliency of network resources and services to hardware and software compent failures is key to a successful high availability strategy in a data center network. Redundancy measures the effects of various failure scenarios on Layer 4-7 services and hardware.

The following tests were performed:

- CSM Redundancy, page 3-80
- FWSM Redundancy, page 3-78
- SSLM Reset, page 3-82

- HSRP Failover, page 3-84

CSM Redundancy

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 follows:

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-HTTP 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-HTTP 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-HTTP 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-HTTP 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-HTTP 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-HTTP 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

- 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 passed.

FWSM Redundancy

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 test follows:

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-HTTP 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-HTTP 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-HTTP 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-HTTP 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

- We expect the FWSM to replicate flow information from active to standby FWSM.
- We expect the standby FWSM will transition to active state with the failure of the active FWSM.

Results

FWSM Redundancy passed.

SSLM Reset

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.

Test Procedure

The procedure used to perform the SSLM Reset test follows:

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 all
```

Issue 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

- 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 passed.

HSRP Failover

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 test follows:

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

- We expect the mean failover time for HSRP to be less then the default dead time of 10 seconds.
- We expect that when the failed system becomes operational again, it will resume HSRP active status and forward traffic.

Results

HSRP Failover passed.

Service Switch Bundle with SSLM 2.1.11

The following tests verified various functional aspects of the three Service Modules (CSM, FWSM and SSLSM) in a bundled fashion; that is, working together in the same chassis to provide services to data center traffic. Three Service Modules are bundled together in the Aggregation Layer switches in DCa, dca-agg-1 and dca-agg-2.

The following test features were conducted:

SSLM 2.1(11) CSM 4.2(6), FWSM 2.3(3.2)

- CSM/SSLSM Integration, page 3-49
- Redundancy, page 3-54

CSM/SSLSM Integration

CSM/SSLSM integration looks at interoperability capacities of the CSM and SSLSM, in terms of how they work together to handle data traffic.

The following tests were performed:

- Backend SSL, page 3-50
- SSL Sticky, page 3-51
- URL Rewrite, page 3-52

Backend SSL

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 Backend SSL test follows:

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 FARM30 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 720.

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:

```
120 times for 1.HTML
240 times for 2.HTML
360 times for (default)
```

Step 12 Issue the **show mod csm 2 real sfarm farm30 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

- We expect the CSM to correctly load balance SSL traffic.
- We expect the CSM to apply the correct L7 policy on clear text traffic.
- We expect the CSM to be able to stick based on the client cookie.
- We expect the SSLSM to re-encrypt 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

Backend SSL passed.

SSL Sticky

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 SSL Sticky test follows:

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 FARM30 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 30 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 FARM30 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

- We expect the CSM to stick clients to the same real based on SSL Session ID.

Results

SSL Sticky passed.

URL Rewrite

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 test follows:

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 following commands:

```
show ssl-proxy stats service rewrite-test
show ssl-proxy stats url
```

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 From the outside windows client, using the Firefox browser, issue an HTTPS request for https://201.40.40.241/p=10,l=http:/201.40.40.241/urlrewrite_dest.html. This URL will cause the word location in the server response to appear in the first packet and the url to appear in the second packet. Stop the ssldump capture and verify the HTTP 302 redirect location field was rewritten to HTTPS.

Step 16 From the outside windows client start ssldump using the servers private key. Using the IE browser issue an HTTPS request for https://201.40.40.241/p=10,l=http:/201.40.40.241/urlrewrite_dest.html. This URL will cause the url in the server response to be split between two packets. Stop the ssldump capture and verify the HTTP 302 redirect location field was rewritten to HTTPS.

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

- We expect that the SSLM can rewrite the server issued 300 series redirects from HTTP to HTTPS.

Results

URL Rewrite passed.

Redundancy

The resiliency of network resources and services to hardware and software compent failures is key to a successful high availability strategy in a data center network. Redundancy measures the effects of various failure scenarios on Layer 4-7 services and hardware.

The following tests were performed:

- FWSM Redundancy, page 3-54
- CSM Redundancy, page 3-56
- SSLM Reset, page 3-59
- HSRP Failover, page 3-61

FWSM Redundancy

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 test follows:

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 system context on the primary and secondary FSWM to verify the primary FWSM is in active state.

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 farm29 detail
show mod csm 2 real sfarm farm30 detail
show mod csm 2 real sfarm farm1 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 farm29 detail
show mod csm 2 real sfarm farm30 detail
show mod csm 2 real sfarm farm1 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 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 farm29 detail
show mod csm 2 real sfarm farm30 detail
show mod csm 2 real sfarm farm1 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 to verify it is in standby state.

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 to verify it is now active.

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 farm29 detail
show mod csm 2 real sfarm farm30 detail
show mod csm 2 real sfarm farm1 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

- We expect the FWSM to replicate flow information from active to standby FWSM.
- We expect the standby FWSM will transition to active state with the failure of the active FWSM.

Results

FWSM Redundancy passed.

CSM Redundancy

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 follows:

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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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

- 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 passed.

SSLM Reset

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.

Test Procedure

The procedure used to perform the SSLM Reset test follows:

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

- 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 passed.

HSRP Failover

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 test follows:

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 in dcb-ss-1, issue the **show failover** system context command to verify that this FWSM is active.

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 detail
show mod csm 2 real sfarm FARM29 detail
show mod csm 2 real sfarm FARM30 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 on the active CSM to verify the connections are being made:

```
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 detail
show mod csm 2 real sfarm FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 stats
show mod csm 2 conn
```

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 using the **show failover** command in the system context.

Step 15 Issue the following commands on the newly active CSM to verify the connections are being made:

```
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 detail
show mod csm 2 real sfarm FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 stats
show mod csm 2 conn
```

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 FWSM in dcb-ss-2 is still active using the **show failover** command in the system context.

In FWSM 2.x software, there is no preemption capability, so this FWSM will remain active until a manual failback is performed.

Step 20 Issue the following commands on the active CSM in dcb-ss-1 to verify the connections are being made:

```
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 detail
show mod csm 2 real sfarm FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 stats
show mod csm 2 conn
```

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 again active using the **show failover** command in the system context.

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

- We expect the mean failover time for HSRP to be less then the default dead time of 10 seconds.
- We expect that when the failed system becomes operational again, it will resume HSRP active status and forward traffic.

Results

HSRP Failover passed.

Service Switch Bundle with SSLM 3.1.1

The following tests verified various functional aspects of the three Service Modules (CSM, FWSM and SSLSM) in a bundled fashion; that is, working together in the same chassis to provide services to data center traffic. Three Service Modules are bundled together in a pair of switches dedicated to providing services, separate from the Aggregation Layer switches.

The following test features were conducted:

SSLM 3.1(1), CSM 4.2(6), FWSM 2.3(3.2)

CSM/FWSM Integration

CSM/FWSM integration looks at interoperability capacities of the CSM and FWSM, in terms of how they work together to handle data traffic.

The following tests were performed:

Active FTP Through FWSM and CSM

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 test follows:

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-a 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-a 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

- 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 passed.

Passive FTP Through FWSM and CSM

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 test follows:

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-A.

The output of the command shows that the serverfarm that is being used is FARM1-A and that **service = ftp** command is enabled.

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 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.

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-a detail** to verify that the previous GET requests have been load balanced evenly across both servers in serverfarm FARM1-A.

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

- We expect the FWSM to permit Passive FTP on the Inside Interface.
- We expect the FWSM will 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 unacceptable impact on CPU or memory.

Results

Passive FTP Through FWSM and CSM passed.

ICMP to a CSM Layer 3 and Layer 4 Vserver

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 to a CSM Layer 3 and Layer 4 Vserver test follows:

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 log
```

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

- 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 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. 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 test follows:

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 In the FWSM Vlan1101-2101 context, use the **show fixup** 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 STT/TUB 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 FARM1.

Step 9 Issue the **show 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 **show module csm 2 real sfarm farm-dns detail** command to verify that each real server in the serverfarm has made some connections.

Step 12 Issue the **clear module csm 2 counters** and **clear module csm 2 connections** commands on the active CSM to clear connections and counters.

Step 13 Use STT/TUB to send a DNS queries at a rate of roughly 1000/second to vserver VIP-DNS for domain name dcb-penguin2.dcb-dcap.cisco.com.

Step 14 While traffic is running, issue the **show xlate** and **show conn | include most** commands on the Vlan1101-2101 FWSM context to verify the xlate table and number of open connections.

Step 15 Verify the results of the DNS query traffic.

Step 16 Use the **show module csm 2 vserver name vip-dns detail** and **show module csm 2 stats** commands on the active CSM to verify the DNS traffic was properly load balanced.

Counters **Tot matches** and **L4 Load Balanced Decisions** should have roughly the same value. Verify the **Tot matches** counter equals roughly the number of attempts from the test tool.

Step 17 Issue the **clear module csm 2 counters** and **clear module csm 2 connections** commands on the active CSM to clear connections and counters.

Step 18 Use STT/TUB to send a DNS queries at a rate of roughly 1500/second to vserver VIP-DNS for domain name dcb-penguin2.dcb-dcap.cisco.com.

Step 19 While traffic is running, issue the **show xlate** and **show conn | include most** commands on the Vlan1101-2101 FWSM context to verify the xlate table and number of open connections.

Step 20 Verify the results of the DNS query traffic.

Step 21 Use the **show module csm 2 vserver name vip-dns detail** and **show module csm 2 stats** commands on the active CSM to verify the DNS traffic was properly load balanced.

Counters **Tot matches** and **L4 Load Balanced Decisions** should have roughly the same value. Verify the **Tot matches** counter equals roughly the number of attempts from the test tool.

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

- We expect the FWSM to permit DNS traffic to vserver VIP-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 CSM Layer4 SYN Attack

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 test follows:

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 1,000,000 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

- We expect the FWSM to intercept syn packets being sent to the CSM reals by issuing a syn cookie.

- We except 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 passed.

Idle Timeout UDP

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 Idle Timeout UDP test follows:

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 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

- We expect flows that exceed the idle timeout will be cleared from both the CSM and the FWSM.
- We expect the CSM vserver to properly load balance UDP traffic.

Results

Idle Timeout UDP passed.

CSM/SSLSM Integration

CSM/SSLSM integration looks at interoperability capacities of the CSM and SSLSM, in terms of how they work together to handle data traffic.

The following tests were performed:

- Backend SSL, page 3-73
- SSL Sticky, page 3-75
- URL Rewrite, page 3-76

Backend SSL

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 Backend SSL test follows:

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 SSLM's to clear statistics:

```
clear ssl-proxy stats service
clear ssl-proxy stats hdr
clear ssl-proxy stats ssl
```

Step 3 Issue the following commands on all four SSLM's 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 FARM30 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 720.

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:

```
120 times for 1.HTML
240 times for 2.HTML
360 times for (default)
```

Step 12 Issue the **show mod csm 2 real sfarm farm30 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 following commands on dcb-ss-1-sslsm-1 to verify that the conns attempted and conns completed counters have incremented and there are no errors.

```
show ssl-proxy stats service BACKENDCLIENT
```

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

- We expect the CSM to correctly load balance SSL traffic.
- We expect the CSM to apply the correct L7 policy on clear text traffic.

- We expect the CSM to be able to stick based on the client cookie.
- We expect the SSLSM to re-encrypt 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

Backend SSL passed.

SSL Sticky

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 SSL Sticky test follows:

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 FARM30 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 30 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 FARM30 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

- We expect the CSM to stick clients to the same real based on SSL Session ID.

Results

SSL Sticky passed.

URL Rewrite

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 test follows:

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 both 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 following commands:

```
show ssl-proxy stats service rewrite-test
show ssl-proxy stats url
```

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 name ssl-rewrite detail** and **show module csm 2 vserver name 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 From the outside windows client, using the Firefox browser, issue an HTTPS request for https://201.40.40.241/p=10,l=http:/201.40.40.241/urlrewrite_dest.html. This URL will cause the word location in the server response to appear in the first packet and the url to appear in the second packet. Stop the ssldump capture and verify the HTTP 302 redirect location field was rewritten to HTTPS.

Step 16 From the outside windows client start ssldump using the servers private key. Using the IE browser issue an HTTPS request for https://201.40.40.241/p=10,l=http:/201.40.40.241/urlrewrite_dest.html. This URL will cause the url in the server response to be split between two packets. Stop the ssldump capture and verify the HTTP 302 redirect location field was rewritten to HTTPS.

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

- We expect that the SSLM can rewrite the server issued 300 series redirects from HTTP to HTTPS.

Results

URL Rewrite passed.

Redundancy

The Catalyst 6500 series of switches provide excellent reliability and network stability by offering a number of Hardware Redundancy options. Dual Supervisor Modules were tested by command or physical (OIR) failure testing.

The following tests were performed:

- FWSM Redundancy, page 3-78
- CSM Redundancy, page 3-80
- SSLM Reset, page 3-82
- HSRP Failover, page 3-84

FWSM Redundancy

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 FWSM Redundancy test follows:

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 system context on the primary and secondary FSWM to verify the primary FWSM is in active state.

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 farm29 detail
show mod csm 2 real sfarm farm30 detail
show mod csm 2 real sfarm farm1 detail
show mod csm 2 stats
show mod csm 2 conn
```

Step 6 Generate HTTPS traffic to vserver SSL29 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 farm29 detail
show mod csm 2 real sfarm farm30 detail
show mod csm 2 real sfarm farm1 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 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 farm29 detail
show mod csm 2 real sfarm farm30 detail
show mod csm 2 real sfarm farm1 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 to verify it is in standby state.

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 to verify it is now active.

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 farm29 detail
show mod csm 2 real sfarm farm30 detail
show mod csm 2 real sfarm farm1 detail
```

```
show mod csm 2 stats
show mod csm 2 conn
```

Step 19 Wait for 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

- We expect the FWSM to replicate flow information from active to standby FWSM.
- We expect the standby FWSM will transition to active state with the failure of the active FWSM.

Results

FWSM Redundancy passed.

CSM Redundancy

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 test follows:

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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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 FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 real sfarm FARM1 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

- 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 passed.

SSLM Reset

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 SSLM Reset test follows:

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

- 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 passed.

HSRP Failover

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 test follows:

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 in dcb-ss-1, issue the **show failover** system context command to verify that this FWSM is active.

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 detail
- show mod csm 2 real sfarm FARM29 detail
- show mod csm 2 real sfarm FARM30 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 on the active CSM to verify the connections are being made:

```
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 detail
show mod csm 2 real sfarm FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 stats
show mod csm 2 conn
```

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 using the **show failover** command in the system context.

Step 15 Issue the following commands on the newly active CSM to verify the connections are being made:

```
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 detail
show mod csm 2 real sfarm FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 stats
show mod csm 2 conn
```

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 FWSM in dcb-ss-2 is still active using the **show failover** command in the system context.

In FWSM 2.x software, there is no preemption capability, so this FWSM will remain active until a manual failback is performed.

Step 20 Issue the following commands on the active CSM in dcb-ss-1 to verify the connections are being made:

```
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 detail
show mod csm 2 real sfarm FARM29 detail
show mod csm 2 real sfarm FARM30 detail
show mod csm 2 stats
show mod csm 2 conn
```

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 again active using the **show failover** command in the system context.

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

- We expect the mean failover time for HSRP to be less then the default dead time of 10 seconds.
- We expect that when the failed system becomes operational again, it will resume HSRP active status and forward traffic.

Results

HSRP Failover passed.

CHAPTER 4

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(2).

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 4-1 depicts the entire SAN topology, including MDS switches, end devices, and test devices. The MDS switches are mostly MDS9513s with these components:

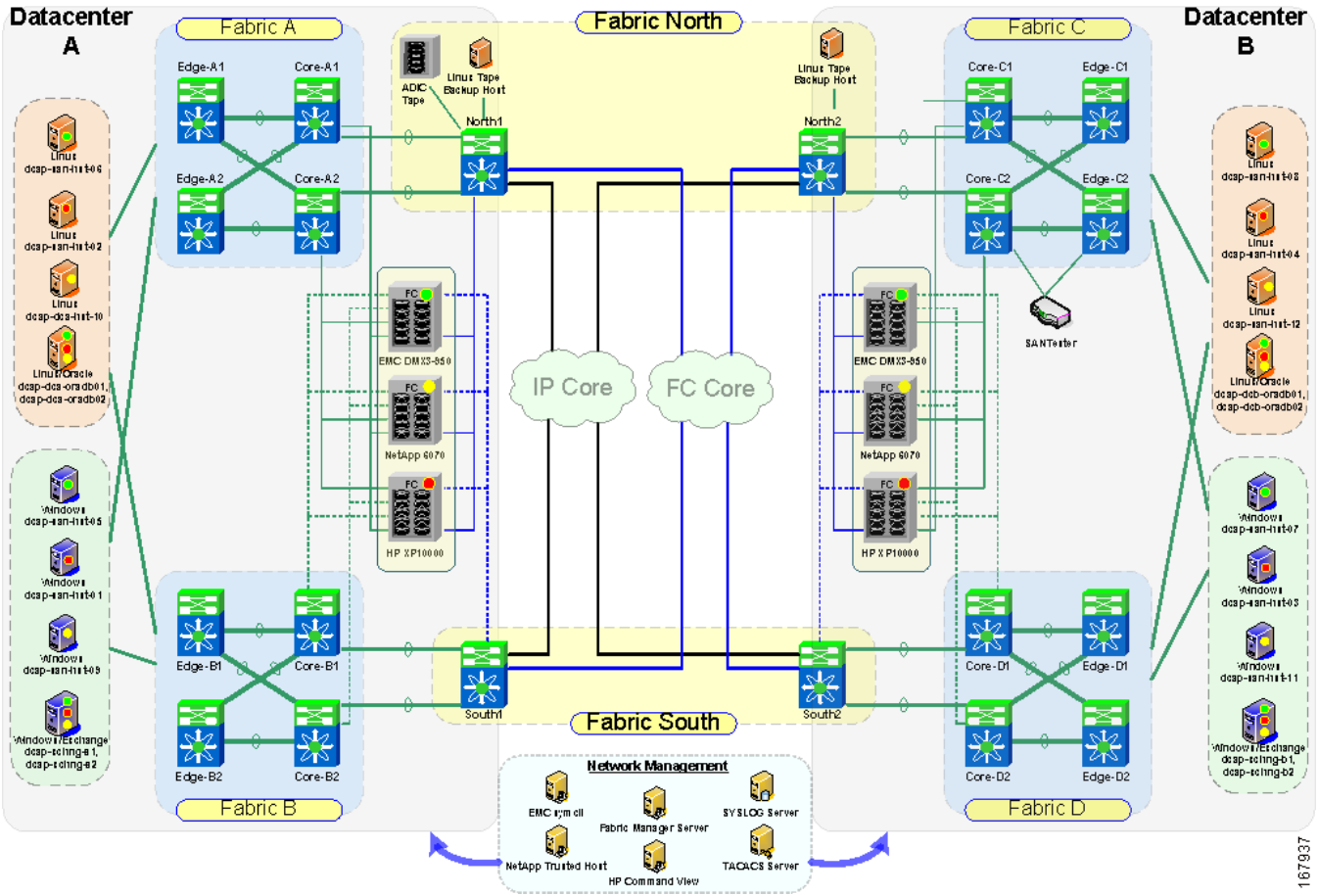
- Redundant version 2 supervisors for high availability and nondisruptive upgrade capability.
- FC modules with 12, 24, and 48 4-Gbps capable ports used for host and storage connectivity as well as interswitch links.
- Storage Service Modules with 32 2-Gbps capable FC ports used for testing replication 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 dual, redundant FC fabrics for connecting hosts to storage as well as dual FC fabrics for synchronous replication and FCIP fabrics for asynchronous storage replication over a transit core consisting of FC SONET links and 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, Hewlett Packard XP10000s, and Network Appliance FAS6070s. The tape library is an 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.

The test devices include an Agilent SAN tester with two 4-port N2X 4 Gbps blades, an Anue Systems SONET delay generator, 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 4-1 DCAP SAN Test Topology Overview



Transport Core

Figure 4-2 shows the infrastructure used by the storage arrays to replicate data between data centers. The FC transit core consists of a pair of Cisco ONS 15454s which provide four STS-24c bidirectional, unprotected circuits over an OC-192 SONET link. Two circuits comprise the north transit fabric and the other two the south transit fabric. All circuits go through an Anue Systems delay generator to allow distance simulation. The circuits in each fabric support native FC traffic for synchronous replication. The IP transit core consists of two access Catalyst 6500 switches and a WAN edge Catalyst 6500 switch per data center. 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 4-3 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 4-4 shows the host and storage ports in Fabric A.

Figure 4-5 shows the host and storage ports in Fabric B.

Figure 4-6 shows the host and storage ports in Fabric C.

Figure 4-7 shows the host and storage ports in Fabric D.

Note The dotted orange lines over dcap-m9509-Core-D1 and dcap-m9509-Edge-D1 indicate only one version 2 supervisor is installed.

Figure 4-8 shows the storage replication ports in the North Transit Fabric.

Figure 4-9 shows the storage replication ports in the South Transit Fabric.

Figure 4-2 DCAP SAN Test Topology Transit Core Detail

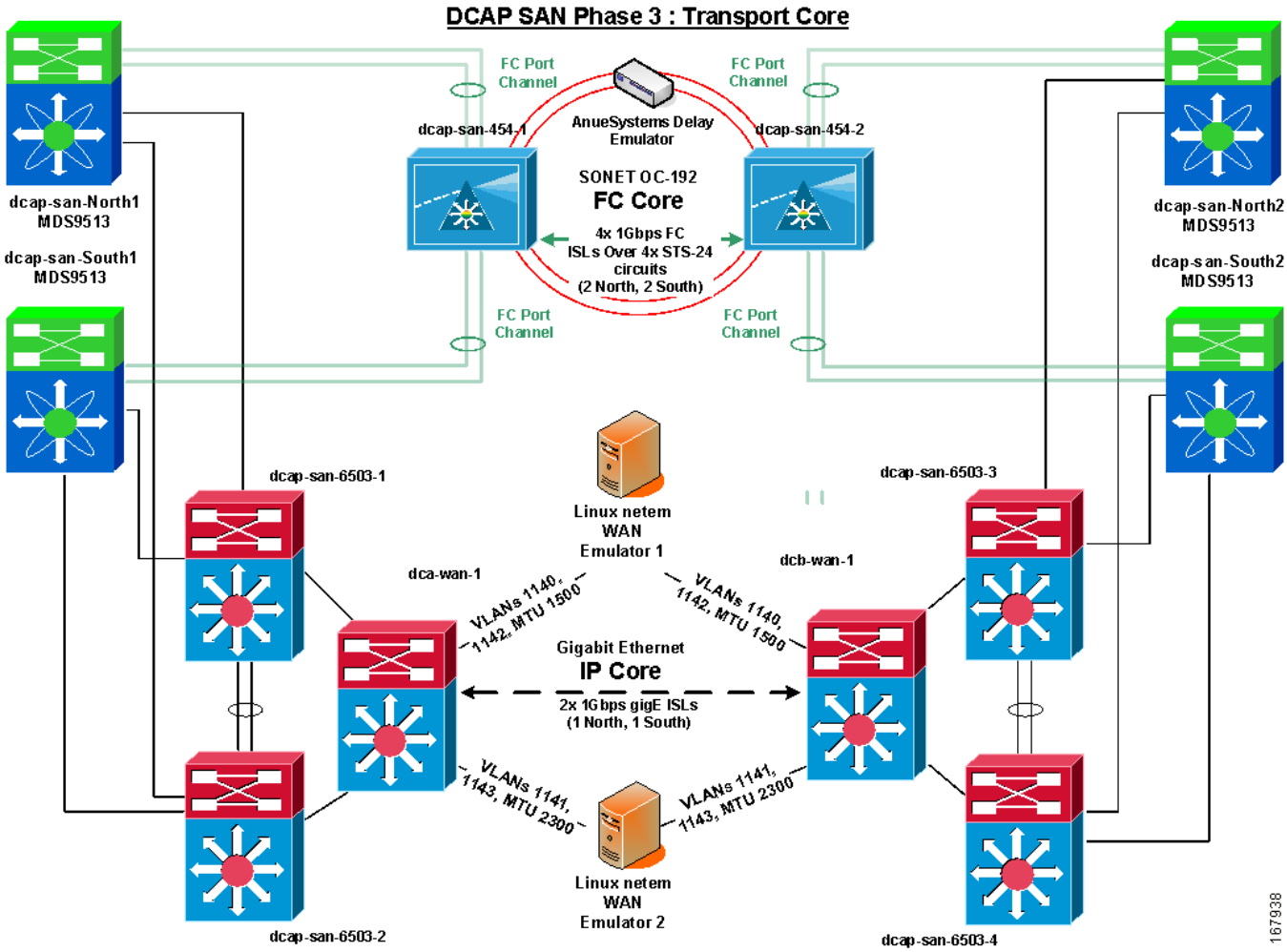


Figure 4-3 DCAP SAN VSAN Data Flow

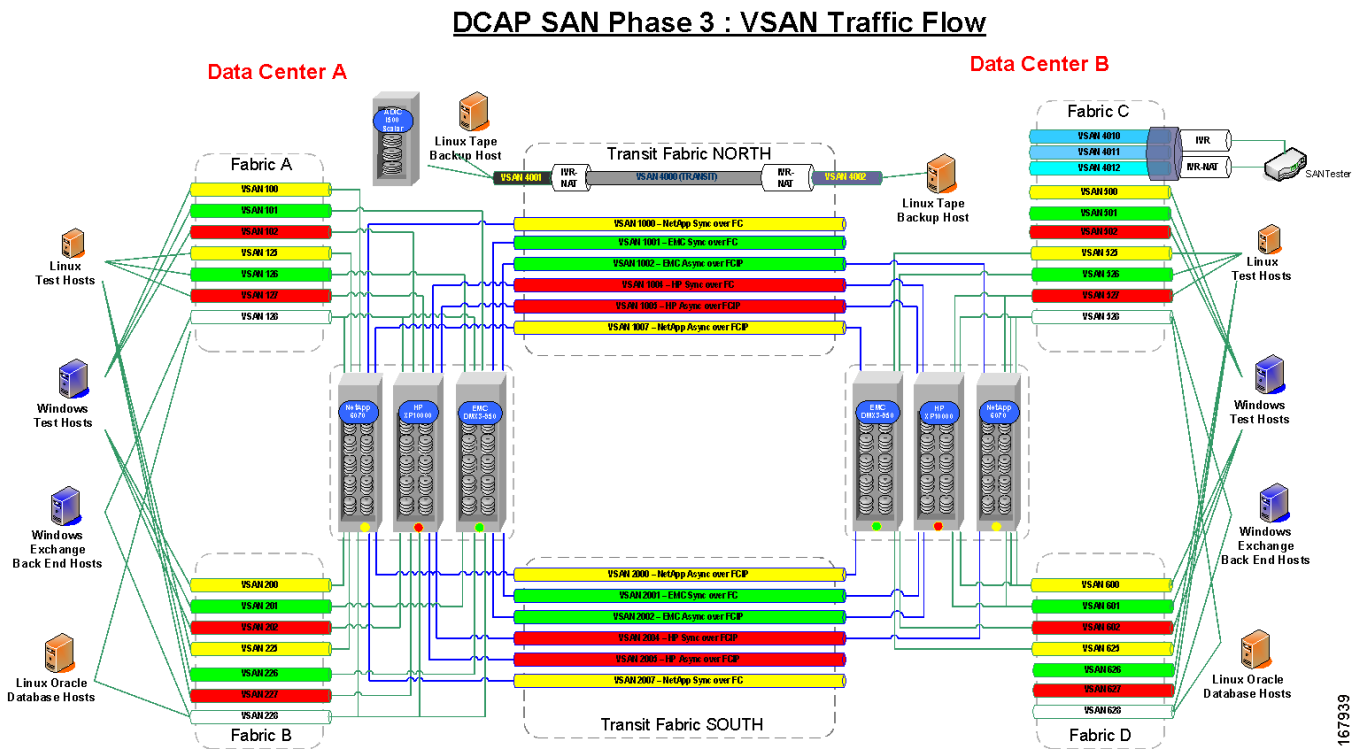


Figure 4-4 Fabric Manager Topology Map for Fabric A

:

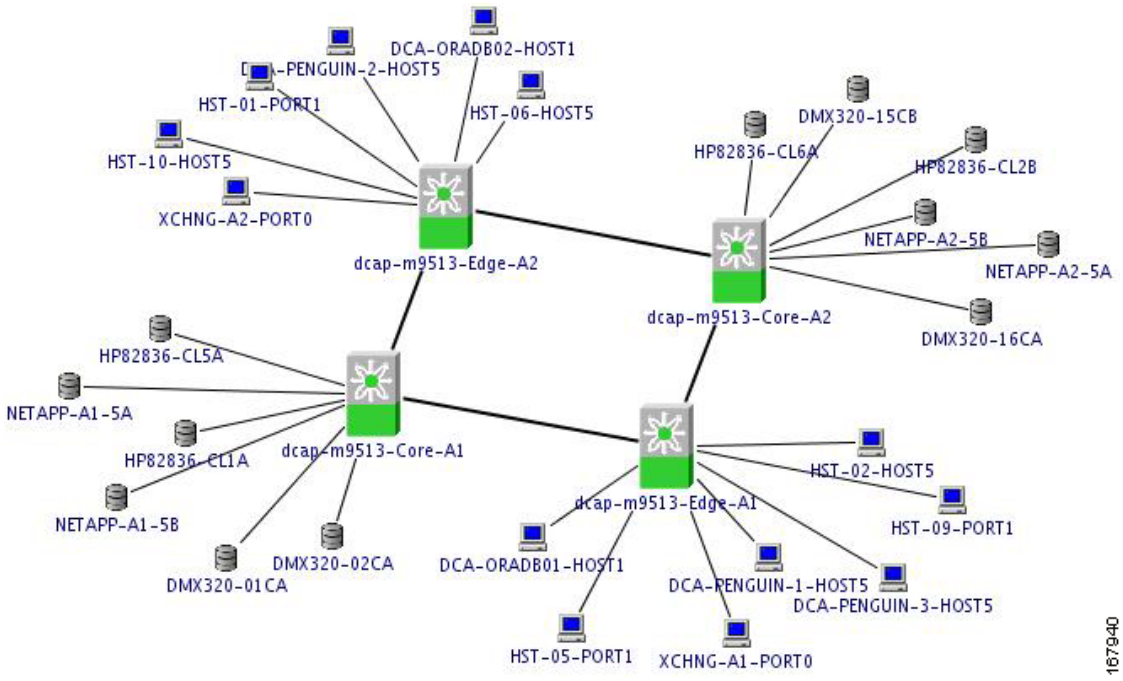


Figure 4-5 Fabric Manager Topology Map for Fabric B

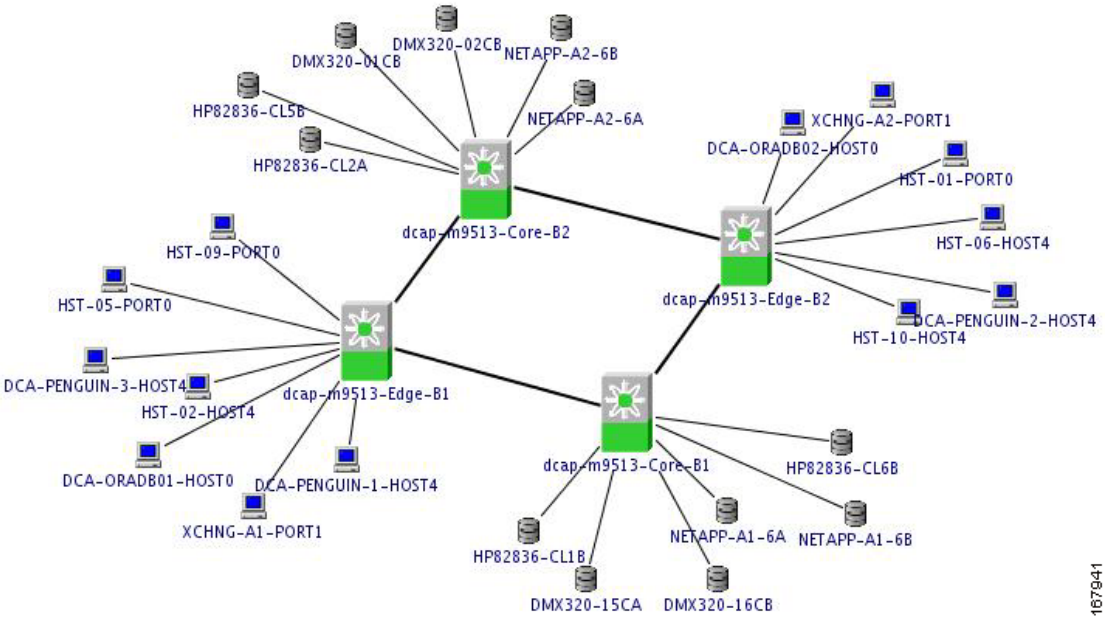


Figure 4-6 Fabric Manager Topology Map for Fabric C

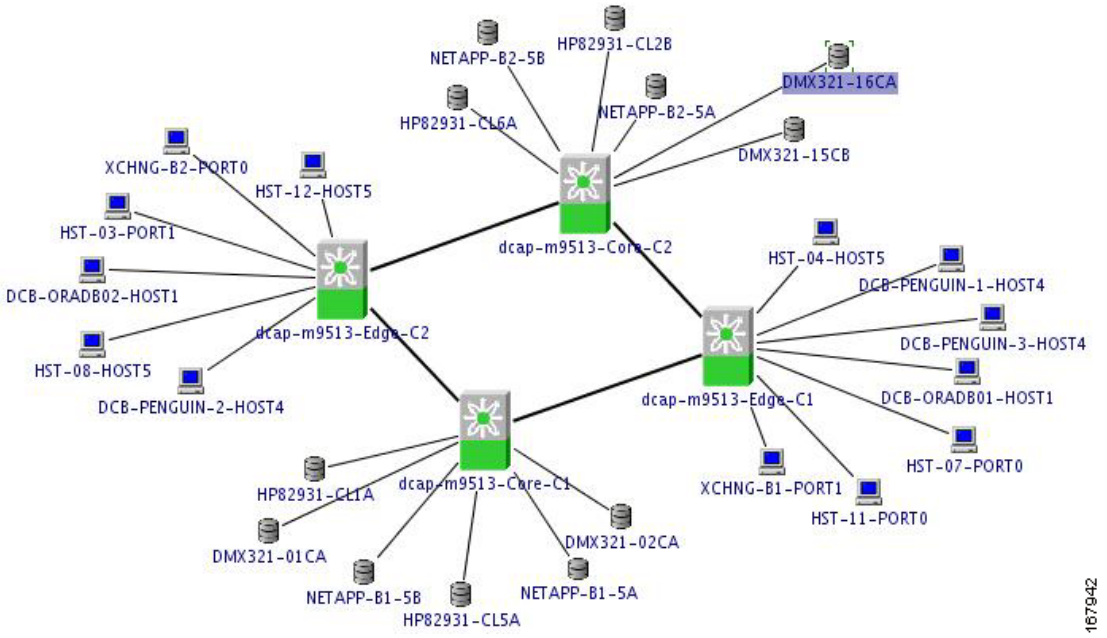


Figure 4-7 Fabric Manager Topology Map for Fabric D

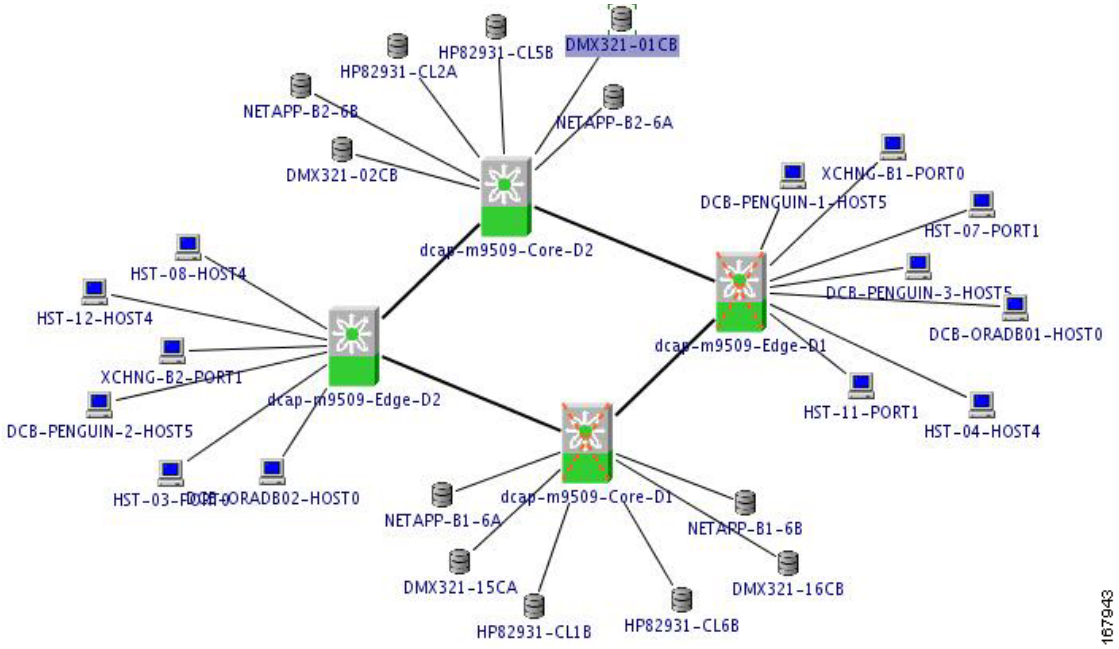


Figure 4-8 Fabric Manager Topology Map for North Transit Fabric

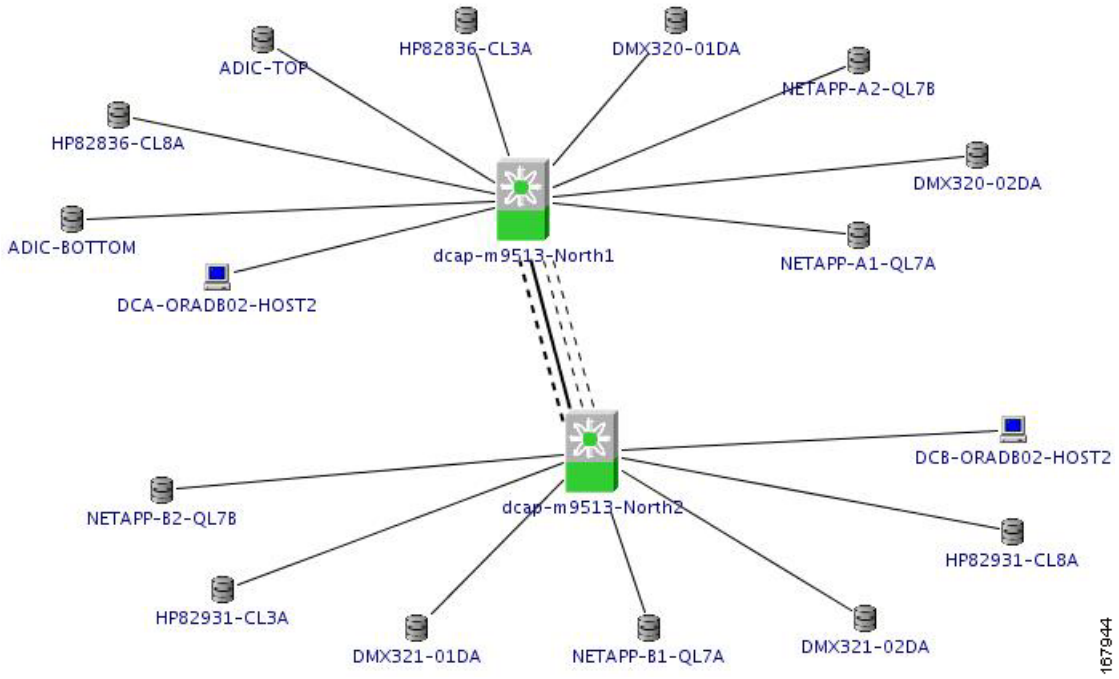
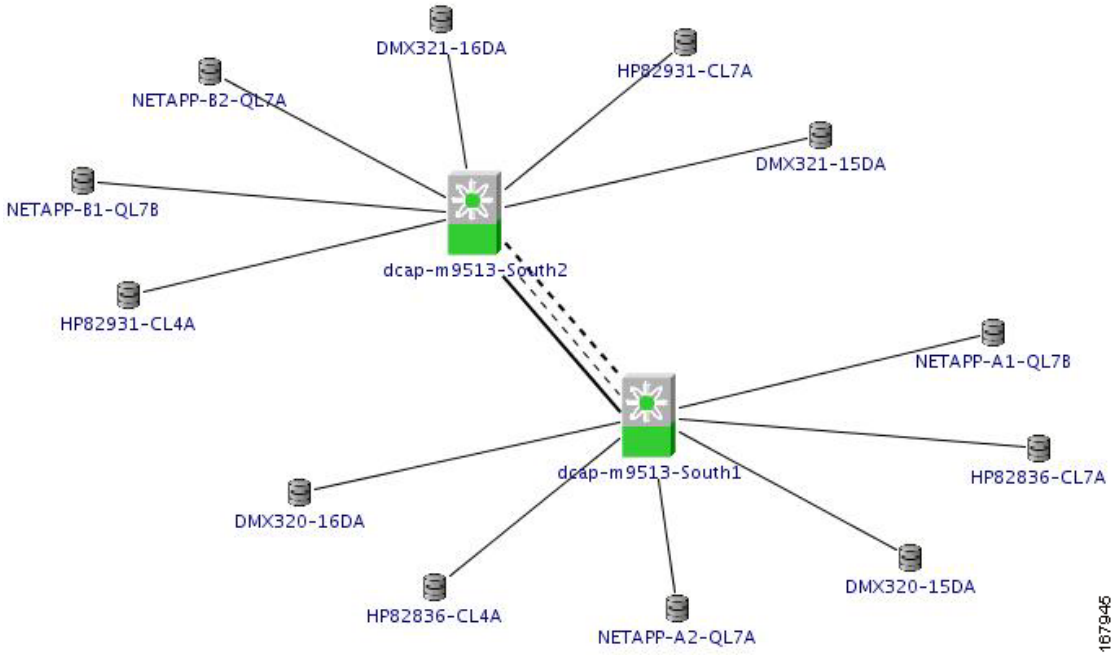


Figure 4-9 Fabric Manager Topology Map for South Transit Fabric



The Cisco DCAP 3.0 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 4-1.

Table 4-1 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

Table 4-1 DCAP SAN Logical Layer Services (continued)

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. Delay generators (Anue Systems for FC, Linux-based netem for FCIP) allow simulation of distance between the data centers. For Phase 3, the tested distance is 100 km (62 miles), which corresponds to a round trip time of 1 ms, for FC and FCIP and both 100 km and 5000 km (80 ms round trip time) for FCIP tape acceleration. 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 a pair of Cisco ONS 15454s which provide four STS-24c bidirectional, unprotected circuits over an OC-192 SONET link. Two circuits comprise the "north" transit fabric and the other two the "south" transit fabric. All circuits go through an Anue Systems delay generator to allow distance simulation. The circuits in each fabric support native FC traffic for synchronous replication. The IP transit core consists of two access Catalyst 6500 switches and a WAN edge Catalyst 6500 switch per data center. 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 MDS9513 (model DS-C9513) with dual version 2 supervisors (model DS-X9530-SF2-K9) for maximum scaleability and nondisruptive operation and upgrades. Host connectivity is through high density 48-port FC modules (model DS-X9148) which provide 4 Gbps of bandwidth on an oversubscribed basis. 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 SONET 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 Anue 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 (model DMX3), Network Appliance (model FAS6070), and Hewlett Packard (model XP10000) as well as an FC tape library from ADIC (model Scalar i500). Each storage frame provides devices for primary application access as well as

replication between data centers. The EMC frames use Synchronous Replication Data Facility/Synchronous (SRDF/S) for synchronous replication and SRDF/Asynchronous (SRDF/A) for asynchronous replication. 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 ADIC 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 and Linux servers with multipath software and either 4-Gbps Qlogic or 2-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 either HP MPIO DSM or Veritas DMP 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 3, both iometer and iorate were configured to do 8 KB sequential writes to generate load.

Test Results Summary

Table 4-2 summarizes tests executed as part of the Cisco DCAP 3.0 testing initiative. Table 4-2 includes the feature or function tested, the section that describes the feature set the feature or function belongs to, the component tests for each feature or function, and whether the test is new in this phase of DCAP testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested. DCAP is designed to cover critical path areas and augment ongoing regression and systems testing.

Table 4-2 Cisco DCAP 3.0 SAN Testing Summary

Test Suites	Features/Functions	Tests	Results
Baseline	A.1: Device Check, page 4-15	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	
	A.2: Infrastructure Check, page 4-19	1. Host and Storage Fabric Connectivity—EMC 2. Host and Storage Fabric Connectivity—NetApp 3. Host and Storage Fabric Connectivity—HP 4. Intra-Fabric Connectivity 5. Topology Discovery—Fabric Manager	CSCsh84608
	A.3: Host to Storage Traffic—EMC, page 4-23	1. Base Setup—VSANs EMC 2. Base Setup—Zoning EMC 3. Host To Storage IO Traffic—EMC 4. Replication FC Sync—EMC 5. Replication FCIP ASync—EMC	
	A.4: Host to Storage Traffic—NetApp, page 4-29	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	
	A.5: Host to Storage Traffic—HP, page 4-34	1. Base Setup—VSANs HP 2. Base Setup—Zoning HP 3. Host To Storage IO Traffic—HP 4. Replication FC-Sync—HP 5. Replication FCIP-ASync—HP 6. Replication FCIP-Async-Journal—HP	

Table 4-2 Cisco DCAP 3.0 SAN Testing Summary (continued)

Test Suites	Features/Functions	Tests	Results
Domain Parameters	Domain Parameters, page 4-41	1. Principal Switch Selection	
FSPF Functionality	FSPF Functionality, page 4-42	1. Basic FSPF Load Balancing 2. Path Selection—Cost Change on Equal Cost Paths 3. Primary Path Failure 4. Primary Path Removal—VSAN Remove	
Fabric Extension	Async Replication—EMC, page 4-46	1. FCIP COMP 100Km EMC 2. FCIP ENCRP 100Km EMC 3. FCIP NONE 100Km EMC 4. FCIP WA 100Km EMC 5. FCIP WA COMP ENCRP 100Km EMC 6. FCIP Portchannel Failure 100Km EMC	
	Async Replication—NetApp, page 4-53	1. FCIP COMP 100Km NETAPP 2. FCIP ENCRP 100Km NETAPP 3. FCIP NONE 100Km NETAPP 4. FCIP WA 100Km NETAPP 5. FCIP WA COMP ENCRP 100Km NETAPP 6. FCIP Portchannel Failure 100Km NETAPP	
	Async Replication—HP, page 4-60	1. FCIP COMP 100Km HP 2. FCIP ENCRP 100Km HP 3. FCIP NONE 100Km HP 4. FCIP WA 100Km HP 5. FCIP WA COMP ENCRP 100Km HP 6. FCIP PortChannel Failure 100Km HP	
	FCIP COMP 100Km HP, page 4-61	1. FC Sync—DST=100Km, WA=OFF - EMC 2. FC Sync—DST=100Km, WA=ON - EMC 3. FC Sync—Portchannel Failure, DST=100Km - EMC	
	Sync Replication—NetApp, page 4-71	1. FC Sync—DST=100Km, WA=OFF - NetApp 2. FC Sync—DST=100Km, WA=ON - NetApp 3. FC Sync—Portchannel Failure, DST=100Km - NetApp	
	Sync Replication—HP, page 4-75	1. FC Sync—DST=100Km, WA=OFF - HP 2. FC Sync—DST=100Km, WA=ON - HP 3. FC Sync—PortChannel Failure, DST=100Km - HP	

Table 4-2 Cisco DCAP 3.0 SAN Testing Summary (continued)

Test Suites	Features/Functions	Tests	Results
Security Functionality	Security Functionality, page 4-79	1. FC SP Authentication Failure 2. Port Security Basic Implementation 3. User Access—TACACS Basic Test 4. User Access—TACACS Servers Failure	
Inter-VSAN Routing Functionality	Inter-VSAN Routing Functionality, page 4-82	1. Basic IVR Implementation 2. Basic IVR-NAT Implementation	
Portchannel Functionality	Portchannel Functionality, page 4-84	1. Basic Portchannel Load Balancing 2. Multiple Link ADD to Group 3. Multiple Links Failure in Group 4. Multiple Links Remove to Group 5. Single Link Add to Group 6. Single Link Failure in Group 7. Single Link Remove from Group	
Resiliency Functionality	EMC, page 4-91	1. Host Link Failure (Link Pull)—EMC 2. Host Link Failure (Port Shutdown)—EMC 3. Host Facing Module Failure (OIR)—EMC 4. Host Facing Module Failure (Reload)—EMC	
	NetApp, page 4-95	1. Host Link Failure (Link Pull)—NETAPP 2. Host Link Failure (Port Shutdown)—NETAPP 3. Host Facing Module Failure (OIR)—NETAPP 4. Host Facing Module Failure (Reload)—NETAPP	
	HP, page 4-99	1. Host Link Failure (Link Pull)—HP 2. Host Link Failure (Port Shutdown)—HP 3. Host Facing Module Failure (OIR)—HP 4. Host Facing Module Failure (Reload)—HP	

Table 4-2 Cisco DCAP 3.0 SAN Testing Summary (continued)

Test Suites	Features/Functions	Tests	Results
Resiliency Functionality	MDS, page 4-102	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 (Power Off) 13. Power Supply Failure (Removal) 14. SAN OS Code Upgrade 15. Standby Supervisor Failure (OIR) 16. Standby Supervisor Failure (Reload) 17. Unused Module Failure (OIR)	

Table 4-2 Cisco DCAP 3.0 SAN Testing Summary (continued)

Test Suites	Features/Functions	Tests	Results
FCIP Tape Acceleration	Tape Read Acceleration, page 4-118	1. Tape Read Acceleration—Local Baseline 2. Tape Read Acceleration—Remote Baseline 3. Tape Read Acceleration—0 km No Compression 4. Tape Read Acceleration—100 km No Compression 5. Tape Read Acceleration—5000 km No Compression 6. Tape Read Acceleration—0 km Hardware Compression 7. Tape Read Acceleration—100 km Hardware Compression 8. Tape Read Acceleration—5000 km Hardware Compression 9. Tape Read Acceleration—0 km Software Compression 10. Tape Read Acceleration—100 km Software Compression 11. Tape Read Acceleration—5000 km Software Compression	
	Tape Write Acceleration, page 4-129	1. Tape Write Acceleration—Local Baseline 2. Tape Write Acceleration—Remote Baseline 3. Tape Write Acceleration—0 km No Compression 4. Tape Write Acceleration—100 km No Compression 5. Tape Write Acceleration—5000 km No Compression 6. Tape Write Acceleration—0 km Hardware Compression 7. Tape Write Acceleration—100 km Hardware Compression 8. Tape Write Acceleration—5000 km Hardware Compression 9. Tape Write Acceleration—0 km Software Compression 10. Tape Write Acceleration—100 km Software Compression 11. Tape Write Acceleration—5000 km Software Compression	

DDTS Summary

Table 4-3 lists Development Defect Tracking System (DDTS) software bugs with descriptions, and comments filed by the DCAP testing team during Cisco DCAP 3.0 SAN testing.

Table 4-3 Summary of DDTS Filed During Cisco DCAP 3.0 SAN Testing

CSCsh84608	Fabric Manager 3.1(2) end to end connectivity tool shows only one fabric

SAN Test Cases

Functionality critical to global enterprises in Cisco DCAP 3.0 Storage Area Network (SAN) testing is described in the following sections. Refer toCisco Data Center Assurance Program (DCAP) 3.0 Configurations document for test device configurations.

Baseline

The baseline tests ensure the SAN topology is configured properly for testing and management tools and devices are operating properly.

The following test features were conducted:

A.1: Device Check

Device check tests ensure management tools and services like the MDS CLI, MDS Fabric Manager, NTP, SMTP, syslog, and so on are accessible and functioning properly.

The following tests were performed:

Device Access—CLI and Device Manager

Individual devices/nodes in the testbed require multiple access methods for management purposes. These access methods include CLI and NMS. 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 [Console] Verify that each node can be accessed and logged into via the console.

Step 3 [Telnet] Verify that each node can be accessed and logged into via Telnet.

Step 4 [SSH] Verify that each node can be accessed and logged into via SSH.

Step 5 [DM] 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

- We expect that access support for console, Telnet, SSH, and Device_Manager is 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 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 Check and 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

- We expect that all linecards or modules are in 'OK' operational condition.
- We expect that all environmental related hardware (e.g., fans, power supply units) are 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 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

- We expect that hardware and environmental status of the nodes can be reviewed and validated via Device_Manager.
- We expect that all linecards or modules are in 'OK' operational condition.
- We expect that all environmental related hardware (e.g., fans, power supply units) are 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 vi NTP, SNMP services for remote management and traps, and logging capabilities to remote SYSLOG servers. This test verified that network services (NTP, SNMP, and SYSLOG) are 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 [NTP] Verify that NTP is configured properly (i.e., peers are the correct NTP servers) and operational (i.e., statistics are working) in all nodes.

Step 3 [NTP] Verify that timestamping is synchronized in all nodes.

Step 4 [SNMP] Verify that SNMP (including traps) is configured properly (i.e., SNMP server IP addresses are correct, traps are enabled, communities set) and operational.

Step 5 [SNMP] Verify that IP connectivity to SNMP servers.

Step 6 [SYSLOG] Verify that SYSLOG (logging server) is configured properly (i.e., SYSLOG server IP addresses are correct, timestamps = milliseconds) and operational.

Step 7 [SYSLOG] Verify that IP connectivity to SYSLOG servers.

Step 8 [SYSLOG] Verify SYSLOG functionality by getting in 'configuration mode' and out. 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

- We expect that NTP services are configured and operational in each node within the testbed.
- We expect that SNMP services are configured and operational in each node within the testbed.
- We expect that SYSLOG services are configured and operational in each node within the testbed.

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, and SYSLOG) are 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 [NTP] Verify that NTP is configured properly (i.e., peers are the correct NTP servers) and operational (i.e., statistics are working) in all nodes.

Step 3 [NTP] Verify that timestamping is synchronized in all nodes.

Step 4 [SNMP] Verify that SNMP (and traps) is configured properly (i.e., SNMP server IP addresses are correct, traps are enabled, communities set) and operational.

Step 5 [SNMP] Verify trap-generation functionality by checking for recent Fabric Manager events. If there aren't any, try generating an SNMP authentication failure. This will generate a trap - check events in FM or DM.

Step 6 [SYSLOG] Verify that SYSLOG (logging server) is configured properly (i.e., SYSLOG server IP addresses are correct, timestamps = milliseconds) and operational.

Step 7 [SYSLOG] Verify IP connectivity to SYSLOG servers.

Step 8 [SYSLOG] 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

- We expect that Device_Manager can accurately verify the active operational state of the services in question.
- We expect that NTP services are configured and operational in each node within the testbed.
- We expect that SNMP services are configured and operational in each node within the testbed.
- We expect that SYSLOG services are configured and operational in each node within the testbed.

Results

Device Network Services Check—Device Manager passed.

A.2: Infrastructure Check

Infrastructure check tests ensures all hosts and storage devices are logged into the appropriate fabric, storage replication works, and Fabric Manager properly discovers all fabrics and reports no anomalies.

The following tests were performed:

- Host and Storage Fabric Connectivity—EMC, page 4-20
- Host and Storage Fabric Connectivity—NetApp, page 4-20
- Host and Storage Fabric Connectivity—HP, page 4-21
- Intra-Fabric Connectivity, page 4-22
- Topology Discovery—Fabric Manager, page 4-23

Host and Storage Fabric Connectivity—EMC

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-hommed 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—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 [FM] Verify that all test host and storage array connections are active and displayed correctly in the topology map.

Step 3 [FM] Verify successful fabric logins, fcns registration, and Device Aliases by checking correct PWWN, HBAs' IDs, aliases.

Step 4 [FM] Check all hosts and storage arrays fabric ports against errors.

Step 5 [CLI] 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

- 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-in into the Fabrics (flogi).

Results

Host and Storage Fabric Connectivity—EMC 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 (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—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 [FM] Verify that all test hosts and storage arrays' connections are active and displayed correctly in the topology map.

Step 3 [FM] Verify successful fabric logins, fcns registration, and Device Aliases by checking correct PWWN, HBAs' IDs, aliases.

Step 4 [FM] Check all hosts and storage arrays fabric ports against errors.

Step 5 [CLI] 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

- We expect that Fabric_Manager's fabric discovery process accurately presents 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-in into the Fabrics (flogi).
- We expect no CPU or memory problems.

Results

Host and Storage Fabric Connectivity—NetApp passed.

Host and Storage Fabric Connectivity—HP

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—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 [FM] Verify that all test hosts and storage arrays' connections are active and displayed correctly in the topology map.

Step 3 [FM] Verify successful fabric logins, fcns registration, and Device Aliases by checking correct PWWN, HBAs' IDs, aliases.

Step 4 [FM] Check all hosts and storage arrays fabric ports against errors.

Step 5 [CLI] 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

- We expect that Fabric Manager's fabric discovery process accurately presents 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 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 passed.

Intra-Fabric Connectivity

Intra-Fabrics' connections (e.g., links, portchannels) 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 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 [FM] Verify that all links between Fabric nodes are active and operational.

Step 3 [FM] Verify that all portchannels are active (i.e., all intended members are active in the channel).

Step 4 [FM]Verify connectivity within the transport fabrics (North and South) - over IP and over SONET. (Screendump from only one VSAN shown.)

Step 5 [CLI] Validate/confirm, via CLI, all intra-Fabric connectivity as verified with FM in the previous steps. Run "show portchannel summary" 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

- We expect that Fabric_Manager successfully discovers all intra-Fabric links/connections with accurate status.
- We expect that all links between fabric nodes are active and operating without problems (e.g., no errors).

- We expect for all portchannels to be fully operational.
- We expect for all fabric nodes to be able to see each other.

Results

Intra-Fabric Connectivity passed with exception CSCsh84608.

Topology Discovery—Fabric Manager

This test verified that Fabric Manager and Device Manager can accurately and completely discover all 6 Fabrics and devices attached to them. The appropriate "Host and Storage Fabric Connectivity" test cases and the "Intra-Fabric Connectivity" test case must have been executed before this one.

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, 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

- We expect the Fabric Manager to be able to discover all nodes in all 6 fabrics without problems.
- We expect no CPU or memory problems.

Results

Topology Discovery—Fabric Manager passed.

A.3: Host to Storage Traffic—EMC

Host-to-storage traffic tests for EMC ensure hosts can access storage devices and that both SRDF/S for synchronous replication and SRDF/A for asynchronous replication are working properly.

The following tests were performed:

- Base Setup—VSANs EMC, page 4-24
- Base Setup—Zoning EMC, page 4-25
- Host To Storage IO Traffic—EMC, page 4-26
- Replication FC Sync—EMC, page 4-27

- Replication FCIP ASync—EMC, page 4-28

Base Setup—VSANs EMC

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 ports 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 (w/ multiple operating systems), storage arrays, and storage array pair. VSANs were configured and verified via Fabric_Manager (w/ CLI validation).

Test Procedure

The procedure used to perform the Base Setup—VSANs 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 [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

- 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 portchannel / trunk ISLs / Fabric Extension links.

Results

Base Setup—VSANs EMC passed.

Base Setup—Zoning EMC

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 ports membership, zone membership, zoneset activation, LUN masking, etc. This test verified the base Zoning configuration to enable communication between hosts (w/ multiple operating systems) and a storage arrays and between storage array pairs. Zones and Zonesets were configured and verified via Fabric_Manager (w/ CLI validation).

Test Procedure

The procedure used to perform the Base Setup—Zoning 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 [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).

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 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 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

- We expect that Fabric_Manager is 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 EMC passed.

Host To Storage IO Traffic—EMC

Host-to-Storage communications is based on IOs where the host reads from and writes to the LUNs in the storage array. This test verified the communication (i.e., I/Os) between hosts (w/ multiple operating systems) and a storage array. Traffic is generated 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 the CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- 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 test follows:

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 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

- We expect complete transport and delivery of all IO traffic between Test-Hosts and Storage-Array
- We expect for Fabric_Manager and the CLI to be able to present accurate link utilization.
- We expect a logical distribution between Read/Write ratios vs. IOPS.

Results

Host To Storage IO Traffic—EMC passed.

Replication FC Sync—EMC

Replication (synchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored real-time (IO is not complete until R2 acknowledges it). This test verified the basic functionality of sync replication between hosts (w/ multiple operating systems) and a storage array pair. Traffic is generated in multiple block sizes with tools like IOMETER and IORATE. Different Read/Write ratios were used. 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- 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 test follows:

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 - VSAN's 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 LUN's 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and 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-sync due to MDS related issues.
- We expect the IO delay statistics to be higher (i.e., longer delay) and less IOPS than the Host-To-Storage scenario.

Results

Replication FC Sync—EMC passed.

Replication FCIP ASync—EMC

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (w/ multiple operating systems) and a storage array pair. Traffic is generated in multiple block sizes 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- 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 test follows:

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. [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 LUN's 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and 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 fail their replication procedures due to MDS related issues.
- We expect the IO delay statistics to be similar (i.e., similar delay) and IOPS to the Host-To-Storage scenario.

Results

Replication FCIP ASync—EMC passed.

A.4: Host to Storage Traffic—NetApp

Host-to-storage traffic tests for NetApp ensure hosts can access storage devices and that synchronous SnapMirror for synchronous replication and asynchronous SnapMirror for asynchronous replication are working properly.

The following tests were performed:

- Base Setup—VSANs NetApp, page 4-29
- Base Setup—Zoning NetApp, page 4-30
- Host To Storage IO Traffic—NetApp, page 4-31
- Replication FC-Sync—NetApp, page 4-32
- Replication FCIP-Async—NetApp, page 4-33

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 ports 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 (w/ multiple operating systems), storage arrays, and storage array pair. VSANs were configured and verified via Fabric_Manager (w/ CLI validation).

- 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 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 [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

- 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 portchannel / trunk ISLs / Fabric Extension links.
- We expect no CPU or memory problems.

Results

Base Setup—VSANs NetApp passed.

Base Setup—Zoning 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 ports membership, zone membership, zoneset activation, LUN masking, etc. This test verified the base Zoning configuration to enable communication between hosts (w/ multiple operating systems) and a storage arrays and between storage array pairs. Zones and Zonesets were configured and verified via Fabric_Manager (w/ 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 - 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).

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

- We expect that Fabric_Manager is 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.
- We expect no CPU or memory problems.

Results

Base Setup—Zoning NetApp passed.

Host To Storage IO Traffic—NetApp

Host-to-Storage communication is based on IOs where the host reads from and writes to the LUNs in the storage array. This test verified the communication (i.e., IOs) between hosts (w/ multiple operating systems) and a storage array. Traffic is 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 CLI (with FM validation) and the test tools. Traffic Generation Characteristics:

- 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 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 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

- We expect complete transport and delivery of all IO traffic between Test-Hosts and Storage-Array.
- We expect the CLI and Fabric_Manager to be able to present accurate link utilization.
- We expect a logical distribution between Read/Write ratios vs. IOPS.
- We expect no CPU or memory problems.

Results

Host To Storage IO Traffic—NetApp passed.

Replication FC-Sync—NetApp

Replication (synchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored real-time (IO is not complete until R2 acknowledges it). This test verified the basic functionality of sync replication between hosts (w/ multiple operating systems) and a storage array pair. 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- 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 VSAN's configuration has been executed and validated in the Base Setup - VSAN's 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 LUN's 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and 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-sync due to MDS related issues.
- We expect the IO delay statistics to be higher (i.e., 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

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (w/ multiple operating systems) and a storage array pair. Traffic is generated in multiple block sizes 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- 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 VSAN's configuration has been executed and validated in the Base Setup - VSAN's 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 LUN's 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and 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 fail their replication procedures due to MDS related issues.
- We expect the IO delay statistics to be similar (i.e., similar delay) and IOPS than the Host-To-Storage scenario.
- We expect no CPU or memory problems.

Results

Replication FCIP-Async—NetApp passed.

A.5: Host to Storage Traffic—HP

The host-to-storage traffic tests for HP ensure hosts can access storage devices and that Continuous Access XP Synchronous for synchronous replication and Continuous Access XP Asynchronous and Journal for asynchronous replication are working properly.

The following tests were performed:

- Base Setup—VSANs HP, page 4-35
- Base Setup—Zoning HP, page 4-36
- Host To Storage IO Traffic—HP, page 4-37
- Replication FC-Sync—HP, page 4-38
- Replication FCIP-ASync—HP, page 4-39

- Replication FCIP-Async-Journal—HP, page 4-40

Base Setup—VSANs HP

.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 (w/ multiple operating systems), storage arrays, and storage array pair. VSANs were configured and verified via Fabric Manager (w/ CLI validation).

Test Procedure

The procedure used to perform the Base Setup—VSANs 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 [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

- 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 portchannel / trunk ISLs / Fabric Extension links.
- We expect no CPU or memory problems.

Results

Base Setup—Zoning HP passed.

Base Setup—Zoning HP

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 - 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).

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

- 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—HP

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 (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 test follows:

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 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

- We expect complete transport and delivery of all IO traffic between test hosts and storage array.
- We expect 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 passed.

Replication FC-Sync—HP

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 test follows:

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 - VSAN's 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 LUN's 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

- 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 passed.

Replication FCIP-ASync—HP

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 (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—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 [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. [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 LUN's 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 LUN's 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

- 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 passed.

Replication FCIP-Async-Journal—HP

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 test follows:

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. [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 LUN's 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

- 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 passed.

Domain Parameters

Domain parameters testing ensures fiber channel domain configuration works as expected.

The following tests were performed:

- Device Access—CLI and Device Manager, page 4-15

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 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

- We expect that hard configuration of Principal Switches across the fabric is successful (no problems or issues).

- We expect that detection, reporting, validation, verification was successfully done with Fabric Manager with confirmation.
- We expect no CPU or memory problems.

Results

Principal Switch Selection passed.

FSPF Functionality

FSPF functionality tests determine whether the Fabric Shortest Path First protocol properly handles load balancing, manual path cost configuration, and path failures and removals.

The following tests were performed:

- Basic FSPF Load Balancing, page 4-42
- Path Selection—Cost Change on Equal Cost Paths, page 4-43
- Primary Path Failure, page 4-44
- Primary Path Removal—VSAN Remove, page 4-44

Basic FSPF Load Balancing

The configuration and verification of FSPF parallel paths load balancing was tested. Redundant parallel paths with equal cost were configured for storage traffic to traverse. All configuration and verification was done via 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 portchannels 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

- We expect that FSPF allow for storage traffic to be successfully load balanced over parallel paths of equal cost.
- We expect that detection, reporting, validation, verification was successfully done with Fabric Manager with CLI confirmation.

Results

Basic FSPF Load Balancing passed.

Path Selection—Cost Change on Equal Cost Paths

The 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 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 portchannels 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

- We expect that FSPF will select a path with better cost than another after cost was change between parallel and equal paths.
- We expect no traffic loss or problems after the path selection has been configured.
- We expect that detection, reporting, validation, verification was successfully done with CLI confirmation.

Results

Path Selection—Cost Change on Equal Cost Paths passed.

Primary Path Failure

This test verified FSPF's detection and re-routing of storage traffic after the primary path is shut down. Storage traffic was generated and traversing a primary path then link is physically disabled. All configuration and verification was done via 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 portchannels 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

- We expect that FSPF will detect the FAILURE(removal) of the primary path and will re-route the traffic over a secondary path.
- We expect no traffic loss or problems after the path selection has been configured.
- We expect that detection, reporting, validation, verification was successfully done with CLI confirmation.

Results

Primary Path Failure passed.

Primary Path Removal—VSAN Remove

This test verified FSPF's detection and re-routing of storage traffic after the VSAN was removed from the primary path. Storage traffic was generated and traversing a primary path - the VSAN was removed from the path. All configuration and verification was done via CLI.

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 portchannels 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

- We expect that FSPF will detect the removal of the primary path and will re-route the traffic over a secondary path.
- We expect no traffic loss or problems after the path selection has been configured.
- We expect that detection, reporting, validation, verification was successfully done with CLI confirmation.

Results

Primary Path Removal—VSAN Remove passed.

Fabric Extension

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 capabilities enabled on the transit fabric.

The following test features were conducted:

- Async Replication—EMC, page 4-46
- Async Replication—NetApp, page 4-53
- Async Replication—HP, page 4-60
- FCIP COMP 100Km HP, page 4-61
- Sync Replication—NetApp, page 4-71
- Sync Replication—HP, page 4-75

Async Replication—EMC

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.

The following tests were performed:

- FCIP COMP 100Km EMC, page 4-46
- FCIP ENCRP 100Km EMC, page 4-47
- FCIP NONE 100Km EMC, page 4-48
- FCIP WA 100Km EMC, page 4-49
- FCIP WA COMP ENCRP 100Km EMC, page 4-50
- FCIP Portchannel Failure 100Km EMC, page 4-52

FCIP COMP 100Km EMC

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (with multiple operating systems) and a storage array pair with the compression feature active in the MDS'es. Traffic is 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : ON
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP COMP 100Km 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 [Pretest Condition 1] Base VSAN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and 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 fail their replication procedures due to MDS related issues.
- We expect the IO BW statistics to improve with the use of the Compression feature.
- We expect no CPU or memory problems.

Results

FCIP COMP 100Km EMC passed.

FCIP ENCRP 100Km EMC

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (with multiple operating systems) and a storage array pair with the IP-Encryption feature active in the MDS'es. Traffic is 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP ENCRP 100Km 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 [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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fall their replication procedures due to MDS related issues.
- We expect the IP-Encryption feature not to have an adverse effect on the IO statistics.
- We expect no CPU or memory problems.

Results

FCIP ENCRP 100Km EMC passed.

FCIP NONE 100Km EMC

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). 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. 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP NONE 100Km 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 [Pretest Condition 1] Base VSAN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 delay statistics to be similar (i.e., similar delay) and IOPS than the Host-To-Storage scenario.
- We expect no CPU or memory problems.

Results

FCIP NONE 100Km EMC passed.

FCIP WA 100Km EMC

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (w/ multiple operating systems) and a storage array pair with the Write-Acceleration feature active in the MDS'es. Traffic is 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 (w/ CLI validation) and the test tools. Traffic Generation Characteristics:

- 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 WA 100Km 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 [Pretest Condition 1] Base VSAN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fall their replication procedures due to MDS related issues.
- We expect the IO delay statistics to improve with the use of the FCIP Write Acceleration feature.
- We expect no CPU or memory problems.

Results

FCIP WA 100Km EMC passed.

FCIP WA COMP ENCRP 100Km EMC

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (with multiple operating systems) and a storage array pair with Write-Acceleration, Compression, and IP-Encryption features active in the MDS'es. Traffic is 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- 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 WA COMP ENCRP 100Km 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 [Pretest Condition 1] Base VSAN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 delay statistics to improve with the use of the FCIP Write Acceleration feature.
- We expect the IO BW statistics to improve with the use of the Compression feature.
- We expect the IP-Encryption feature not to have an adverse effect on the IO statistics.
- We expect the combination of WA, Compression, and Encryption not to have an negative effect on FCIPs functionality or the storage traffic delivery.
- We expect no CPU or memory problems.

Results

FCIP WA COMP ENCRP 100Km EMC passed.

FCIP Portchannel Failure 100Km EMC

This test verified the resilience of the Fabric Extension network (over IP) when one portchannel link fails while Async replication is active. The traffic statistics (IO Delay, IO 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
- Write-Acceleration : ON
- Compression : ON
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP Portchannel Failure 100Km 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 [Pretest Condition 1] Base VSAN's 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 portchannel towards the transport (IP fabric extension) fabric.

Step 9 Confirm that the portchannel 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 portchannel link.

Step 11 Re-establish the portchannel failed link.

Step 12 Verify failed link is reestablished as a member of the portchannel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the portchannel 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fail Async replication due to MDS related issues like the Fabric Extension portchannel failure.
- We expect the portchannel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FCIP Portchannel Failure 100Km EMC passed.

Async Replication—NetApp

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.

The following tests were performed:

- FCIP COMP 100Km NETAPP, page 4-53
- FCIP ENCRP 100Km NETAPP, page 4-54
- FCIP NONE 100Km NETAPP, page 4-56
- FCIP WA 100Km NETAPP, page 4-57
- FCIP WA COMP ENCRP 100Km NETAPP, page 4-58
- FCIP Portchannel Failure 100Km NETAPP, page 4-59

FCIP COMP 100Km NETAPP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (with multiple operating systems) and a storage array pair with the Compression feature active in the MDS'es. Traffic is 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 (w/ CLI validation) and the test tools. Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 4K, 8K, 16K, 32K
- Write-Acceleration : OFF

- Compression : ON
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP COMP 100Km 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's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fall their replication procedures due to MDS related issues.
- We expect the IO BW statistics to improve with the use of the Compression feature.
- We expect no CPU or memory problems.

Results

FCIP COMP 100Km NETAPP passed.

FCIP ENCRP 100Km NETAPP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (with multiple operating systems) and a storage array pair with the IP-Encryption feature active in the MDS'es. Traffic is 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP ENCRP 100Km 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's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fall their replication procedures due to MDS related issues.
- We expect the IP-Encryption feature not to have an adverse effect on the IO statistics.
- We expect no CPU or memory problems.

Results

FCIP ENCRP 100Km NETAPP passed.

FCIP NONE 100Km NETAPP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). 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. 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP NONE 100Km 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's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager and CLI 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 delay statistics to be similar (i.e., similar delay) and IOPS than the Host-To-Storage scenario.

- We expect no CPU or memory problems.

Results

FCIP NONE 100Km NETAPP passed.

FCIP WA 100Km NETAPP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (w/ multiple operating systems) and a storage array pair with the Write-Acceleration feature active in the MDS'es. Traffic is 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 (w/ CLI validation) and the test tools.
Traffic Generation Characteristics:

- 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 WA 100Km 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's 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fall their replication procedures due to MDS related issues.
- We expect the IO delay statistics to improve with the use of the FCIP Write Acceleration feature.
- We expect no CPU or memory problems.

Results

FCIP WA 100Km NETAPP passed.

FCIP WA COMP ENCRP 100Km NETAPP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (w/ multiple operating systems) and a storage array pair with Write-Acceleration, Compression, and IP-Encryption features active in the MDS'es. Traffic is 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- 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 WA COMP ENCRP 100Km 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's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 delay statistics to improve with the use of the FCIP Write Acceleration feature.
- We expect the IO BW statistics to improve with the use of the Compression feature.
- We expect the IP-Encryption feature not to have an adverse effect on the IO statistics.
- We expect the combination of WA, Compression, and Encryption not to have an negative effect on FCIPs functionality or the storage traffic delivery.
- We expect no CPU or memory problems.

Results

FCIP WA COMP ENCRP 100Km NETAPP passed.

FCIP Portchannel Failure 100Km NETAPP

This test verified the resilience of the Fabric Extension network (over IP) when one portchannel link fails while Async replication is active. The traffic statistics (IO Delay, IO 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
- Write-Acceleration : ON
- Compression : ON
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP Portchannel Failure 100Km 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's 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 portchannel towards the transport (IP fabric extension) fabric.

Step 9 Confirm that the portchannel 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 portchannel link.

Step 11 Reestablish the portchannel failed link.

Step 12 Verify failed link is reestablished as a member of the portchannel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the portchannel 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fail Async replication due to MDS related issues like the Fabric Extension port-channal failure.
- We expect the portchannel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FCIP Portchannel Failure 100Km NETAPP passed.

Async Replication—HP

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.

- FCIP COMP 100Km HP, page 4-61

FCIP COMP 100Km HP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (with multiple operating systems) and a storage array pair with the Compression feature active in the MDS'es. Traffic is 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 (w/ CLI validation) and the test tools.
Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 4K, 8K, 16K, 32K
- Write-Acceleration : OFF
- Compression : ON
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP COMP 100Km 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 [Pretest Condition 1] Base VSAN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fall their replication procedures due to MDS related issues.
- We expect the IO BW statistics to improve with the use of the Compression feature.
- We expect no CPU or memory problems.

Results

FCIP COMP 100Km HP passed.

FCIP ENCRP 100Km HP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (with multiple operating systems) and a storage array pair with the IP-Encryption feature active in the MDS'es. Traffic is 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : ON
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP ENCRP 100Km 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 [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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fall their replication procedures due to MDS related issues.
- We expect the IP-Encryption feature not to have an adverse effect on the IO statistics.
- We expect no CPU or memory problems.

Results

FCIP ENCRP 100Km HP passed.

FCIP NONE 100Km HP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). 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. 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- Read/Write ratios : 0/100
- Block Sizes per r/w ratio : 8K
- Write-Acceleration : OFF
- Compression : OFF
- Encryption : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FCIP NONE 100Km 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 [Pretest Condition 1] Base VSAN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager and CLI 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 delay statistics to be similar (i.e., similar delay) and IOPS than the Host-To-Storage scenario.
- We expect no CPU or memory problems.

Results

FCIP NONE 100Km HP passed.

FCIP WA 100Km HP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (w/ multiple operating systems) and a storage array pair with the Write-Acceleration feature active in the MDS'es. Traffic is 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 (w/ CLI validation) and the test tools.
Traffic Generation Characteristics:

- 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 WA 100Km 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 [Pretest Condition 1] Base VSAN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for Fabric_Manager to be able to present accurate link utilization.
- We expect the Storage-Array pair not to fall their replication procedures due to MDS related issues.
- We expect the IO delay statistics to improve with the use of the FCIP Write Acceleration feature.
- We expect no CPU or memory problems.

Results

FCIP WA 100Km HP passed.

FCIP WA COMP ENCRP 100Km HP

Replication (Asynchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored in time offsets (IO is not complete after R1 acknowledges it but cached data is replicated to R2 in time intervals). This test verified the basic functionality of Async replication between hosts (w/ multiple operating systems) and a storage array pair with Write-Acceleration, Compression, and IP-Encryption features active in the MDS'es. Traffic is 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 CLI (with Fabric_Manager validation) and the test tools. Traffic Generation Characteristics:

- 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 WA COMP ENCRP 100Km 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 [Pretest Condition 1] Base VSAN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 delay statistics to improve with the use of the FCIP Write Acceleration feature.
- We expect the IO BW statistics to improve with the use of the Compression feature.
- We expect the IP-Encryption feature not to have an adverse effect on the IO statistics.
- We expect the combination of WA, Compression, and Encryption not to have an negative effect on FCIPs functionality or the storage traffic delivery.
- We expect no CPU or memory problems.

Results

FCIP WA COMP ENCRP 100Km HP passed.

FCIP PortChannel Failure 100Km HP

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 PortChannel Failure 100Km 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 [Pretest Condition 1] Base VSAN's 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 portchannel towards the transport (IP fabric extension) fabric.

Step 9 Confirm that the portchannel 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 portchannel link.

Step 11 Reestablish the portchannel failed link.

Step 12 Verify failed link is reestablished as a member of the portchannel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the portchannel 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

- 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 portchannel failure.
- We expect the portchannel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FCIP PortChannel Failure 100Km HP passed.

Sync Replication—EMC

Synchronous replication test for EMC tests SRDF/S with and without FC write acceleration.

The following tests were performed:

- FC Sync—DST=100Km, WA=OFF - EMC, page 4-68
- FC Sync—DST=100Km, WA=ON - EMC, page 4-69
- FC Sync—Portchannel Failure, DST=100Km - EMC, page 4-70

FC Sync—DST=100Km, WA=OFF - EMC

Replication (synchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored real-time (IO is not complete until R2 acknowledges it). This test verified the basic functionality of sync replication between hosts (w/ multiple operating systems) and a storage array pair. Traffic is 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 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 : OFF
- Iteration time : 5 minutes
- Distance : 100 Km

Test Procedure

The procedure used to perform the FC Sync—DST=100Km, WA=OFF - 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 [Pretest Condition 1] Base VSAN's configuration has been executed and validated in the Base Setup - VSAN's 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 LUN's 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 (i.e., longer delay) and less IOPS than the Host-To-Storage scenario.
- We expect no CPU or memory problems.

Results

FC Sync—DST=100Km, WA=OFF - EMC passed.

FC Sync—DST=100Km, WA=ON - EMC

Replication (synchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored real-time (IO is not complete until R2 acknowledges it). This test verified the basic functionality of sync replication between hosts (with multiple operating systems) and a storage array pair with the Write-Acceleration feature active. Traffic is 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 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 Sync—DST=100Km, WA=ON - 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 [Pretest Condition 1] Base VSAN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 lower than the non-FCWA scenario at the same distance.
- We expect no CPU or memory problems.

Results

FC Sync—DST=100Km, WA=ON - EMC passed.

FC Sync—Portchannel Failure, DST=100Km - EMC

This test verified the resilience of the Fabric Extension network when one portchannel link fails while sync replication is active. The traffic statistics (IO Delay, IO per second) were observed, validated, and collected by Fabric_Manager (w/ CLI 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 Sync—Portchannel Failure, DST=100Km - 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 [Pretest Condition 1] Base VSAN's 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 LUN's without problems or issues.

Step 7 Verify FCWA statistics to show the feature is operational.

Step 8 Fail one link off the FC portchannel within the Transport (fabric extension) Fabric.

Step 9 Confirm that the portchannel 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 portchannel link.

Step 11 Re-establish the portchannel failed link.

Step 12 Verify failed link is reestablished as a member of the portchannel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the portchannel 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 like the Fabric Extension port-channal failure.
- We expect the portchannel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FC Sync—Portchannel Failure, DST=100Km - EMC passed.

Sync Replication—NetApp

Synchronous replication test for NetApp tests synchronous SnapMirror with and without FC write acceleration.

The following tests were performed:

- FC Sync—DST=100Km, WA=OFF - NetApp, page 4-72
- FC Sync—DST=100Km, WA=ON - NetApp, page 4-73
- FC Sync—Portchannel Failure, DST=100Km - NetApp, page 4-74

FC Sync—DST=100Km, WA=OFF - NetApp

Replication (synchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored real-time (IO is not complete until R2 acknowledges it). This test verified the basic functionality of sync replication between hosts (with multiple operating systems) and a storage array pair. Traffic is 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 (w/ CLI validation) and the test tools.
Traffic Generation Characteristics:

- 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 Sync—DST=100Km, WA=OFF - 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's 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 LUN's 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 (i.e., longer delay) and less IOPS than the Host-To-Storage scenario.
- We expect no CPU or memory problems.

Results

FC Sync—DST=100Km, WA=OFF - NetApp passed.

FC Sync—DST=100Km, WA=ON - NetApp

Replication (synchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored real-time (IO is not complete until R2 acknowledges it). This test verified the basic functionality of sync replication between hosts (with multiple operating systems) and a storage array pair with the Write-Acceleration feature active. Traffic is 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 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 Sync—DST=100Km, WA=ON - 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.

- We expect for 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 lower than the non-FCWA scenario at the same distance.
- We expect no CPU or memory problems.

Results

FC Sync—DST=100Km, WA=ON - NetApp passed.

FC Sync—Portchannel Failure, DST=100Km - NetApp

This test verified the resilience of the Fabric Extension network when one portchannel link fails while sync replication is active. The traffic statistics (IO Delay, IO per second) were observed, validated, and collected by Fabric_Manager (w/ CLI 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 Sync—Portchannel Failure, DST=100Km - 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's 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 LUN's without problems or issues.

Step 7 Verify FCWA statistics to show the feature is operational.

Step 8 Fail one link off the FC portchannel within the transport (fabric extension) fabric.

Step 9 Confirm that the portchannel 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 portchannel link.

Step 11 Reestablish the portchannel failed link.

Step 12 Verify failed link is reestablished as a member of the portchannel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the portchannel 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.

Step 17

Expected Results

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 like the Fabric Extension port-channal failure.
- We expect the portchannel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FC Sync—Portchannel Failure, DST=100Km - NetApp passed.

Sync Replication—HP

Synchronous replication test for HP tests HP XP Continuous Access Synchronous replication with and without FC write acceleration.

The following tests were performed:

- FC Sync—DST=100Km, WA=OFF - HP, page 4-75
- FC Sync—DST=100Km, WA=ON - HP, page 4-76
- FC Sync—PortChannel Failure, DST=100Km - HP, page 4-77

FC Sync—DST=100Km, WA=OFF - HP

Replication (synchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored real-time (IO is not complete until R2 acknowledges it). This test verified the basic functionality of sync replication between hosts (with multiple operating systems) and a storage array pair. Traffic is 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 (w/ CLI validation) and the test tools. Traffic Generation Characteristics:

- 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 Sync—DST=100Km, WA=OFF - 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 [Pretest Condition 1] Base VSAN's configuration has been executed and validated in the Base Setup - VSAN's 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 LUN's 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 (i.e., longer delay) and less IOPS than the Host-To-Storage scenario.
- We expect no CPU or memory problems.

Results

FC Sync—DST=100Km, WA=OFF - HP passed.

FC Sync—DST=100Km, WA=ON - HP

Replication (synchronous) services is based on IOs from hosts to array-R1 to array-R2 where LUNs are mirrored real-time (IO is not complete until R2 acknowledges it). This test verified the basic functionality of sync replication between hosts (with multiple operating systems) and a storage array pair with the Write-Acceleration feature active. Traffic is 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 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 Sync—DST=100Km, WA=ON - 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 [Pretest Condition 1] Base VSAN's 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 LUN's 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

- We expect complete transport, delivery, and replication of all IO traffic between Test-Hosts and Storage-Array pair for all block size iterations within the Read/Write ratio.
- We expect for 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 lower than the non-FCWA scenario at the same distance.
- We expect no CPU or memory problems.

Results

FC Sync—DST=100Km, WA=ON - HP passed.

FC Sync—PortChannel Failure, DST=100Km - HP

:This test verified the resilience of the fabric extension network when one portchannel 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 Sync—PortChannel Failure, DST=100Km - 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 [Pretest Condition 1] Base VSAN's 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 portchannel within the transport (fabric extension) fabric.

Step 9 Confirm that the portchannel 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 portchannel link.

Step 11 Reestablish the portchannel failed link.

Step 12 Verify failed link is reestablished as a member of the portchannel and that the recovery was detected and reported to the management applications.

Step 13 Verify that replication traffic is load balanced across the portchannel 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

- 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 portchannel failure.
- We expect the portchannel link failure to be detected and reported to the management applications.
- We expect no CPU or memory problems.

Results

FC Sync—PortChannel Failure, DST=100Km - HP passed.

Security Functionality

Security functionality tests checked the proper operation of the FC Security Protocol (FC-SP), port security, and TACACS+ access control.

The following tests were performed:

- FC SP Authentication Failure, page 4-79
- Port Security Basic Implementation, page 4-80
- User Access—TACACS Basic Test, page 4-80
- User Access—TACACS Servers Failure, page 4-81

FC SP Authentication Failure

This test verified FC-SP 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

- We expect that FC-SP successfully rejects the integration of an edge switch with wrong credentials.
- We expect that detection, reporting, validation, verification was successfully done with 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 per port and then replaced with another non-authorized host. All configuration and verification was done via Fabric Manager with confirmation through 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

- We expect that Port-Security is configured without problems or issues.
- We expect for Port-Security to automatically learn the address of the test-host and keep it locked after auto-learn is disabled.
- We expect for Port-Security to reject (or not allow) the incorrect non-authorized node to login into the fabric.
- We expect that detection, reporting, validation, verification was successfully done with 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 authorization/authentication main mechanism in the testbed. Remote (TACACS+) authorization/authentication is validated. All configuration and verification was done via Fabric Manager with confirmation through 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.

step7-test_user_access_tacacs_basic_step_7.csv

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

- We expect successful access and Remote Authorization/Authentication.
- We expect for access to be denied with wrong Username/Password combination.
- We expect that detection, reporting, validation, verification was successfully done with Fabric Manager with 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 last resort in the MDS. All configuration and verification was done via Fabric Manager with confirmation through 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

- We expect the MDS to successfully access the TACACS+ secondary servers if communication to the primary is lost.
- We expect for local authorization to be used as last resort when all TACACS+ servers are down.
- We expect that detection, reporting, validation, verification was successfully done with Fabric Manager with CLI confirmation.
- We expect no CPU or memory problems.

Results

User Access—TACACS Servers Failure passed.

Inter-VSAN Routing Functionality

Inter-VSAN (IVR) routing functionality tests make sure IVR works as expected both with and without NAT.

The following tests were performed:

- Basic IVR Implementation, page 4-82
- Basic IVR-NAT Implementation, page 4-83

Basic IVR Implementation

IVR basic Inter-VSAN routing functionality was tested. Traffic generation generated across the Transport Fabrics (transit) in other VSANs. All configuration done using Fabric Manager and verification was done via 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

- We expect that IVR will successfully route storage traffic from the Source VSAN over the Transit/Transport Fabric (transit vsan) to the remote Fabric (destination).
- We expect no loss of traffic or problems with the Inter-VSAN routing.

Results

Basic IVR Implementation passed.

Basic IVR-NAT Implementation

IVR-NAT basic Inter-VSAN routing functionality was tested between Nodes/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 same Domain-IDs. 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

- We expect that IVR-NAT will successfully route Fiber Channel traffic from the Source VSAN over the 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 that detection, reporting, validation, verification was successfully done with CLI confirmation.

Results

Basic IVR-NAT Implementation passed.

Portchannel Functionality

Portchannel functionality tests look at whether the port channel protocol correctly load balances, allows link additions and removals, and handles link failures.

The following tests were performed:

- Basic Portchannel Load Balancing, page 4-84
- Multiple Link ADD to Group, page 4-85
- Multiple Links Failure in Group, page 4-86
- Multiple Links Remove to Group, page 4-87
- Single Link Add to Group, page 4-88
- Single Link Failure in Group, page 4-89
- Single Link Remove from Group, page 4-89

Basic Portchannel Load Balancing

This test verified the portchannel's load-balancing capability (based on OXID) across the 12xISL inter-fabric channels. Storage traffic was generated to cross the portchannels. All configuration and verification was done via CLI.

Test Procedure

The procedure used to perform the Basic Portchannel 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 12 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 12 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 12 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

- We expect that storage traffic is evenly distributed across all members of the portchannels without loss or problems (based on OXID).
- We expect no CPU or memory problems.

Results

Basic Portchannel Load Balancing passed.

Multiple Link ADD to Group

This test verified the support for the addition of multiple links to an active portchannel group without disrupting active traffic or services over the channel. Storage traffic was generated to cross the portchannels prior, during, and after the links were added. All configuration and verification was done via Fabric Manager with confirmation through 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 6 member links from a port channel (via configuration) between an edge and a core switch. The port channel must have 6 ISL members after removal.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across the port channel with 6 ISL members to a core MDS.

Step 4 Verify storage traffic is flowing without loss or problems over portchannels.

Step 5 Verify that storage traffic traversing the 6 ISL port channel members is evenly distributed across all channel members.

Step 6 Add 6 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 12 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

- We expect that the multiple link addition/integration is executed successfully without errors, issues, or problems.
- We expect that the multiple link addition didn't affect active storage traffic over the portchannel.
- 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 that the physical failure of multiple links in an active portchannel group does not disrupt active traffic or services over the channel. Storage traffic was generated to cross the portchannels prior, during, and after the link failure. All configuration and verification was done via 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 12 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 12 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 12 members.

Step 6 Physically remove six 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

- We expect that a multiple links failure doesn't affect active storage traffic over the portchannel.

- We expect the failure to be detected and reported by the nodes to the management applications/servers.
- We expect the active storage traffic to be load-balanced over the remaining portchannel members.
- We expect no CPU or memory problems.

Results

Multiple Links Failure in Group passed.

Multiple Links Remove to Group

This test verified the support for the removal of multiple links from an active portchannel group without disrupting active traffic or services over the channel. Storage traffic was generated to cross the portchannels prior, during, and after the link was removed. All configuration and verification was done via CLI.

Test Procedure

The procedure used to perform the Multiple Links Remove 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 The port channel under test must be configured and active and contain 12 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 12 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 12 members.

Step 6 Remove 6 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

- We expect that a multiple links removal is executed successfully without errors, issues, or problems.
- We expect that the multiple links removal didn't affect active storage traffic over the portchannel.
- We expect the active storage traffic to be load-balanced over the remaining portchannel members.
- We expect no CPU or memory problems.

Results

Multiple Links Remove to Group passed.

Single Link Add to Group

This test verified the support for the addition of a single link to an active portchannel group without disrupting active traffic or services over the channel. Storage traffic was generated to cross the portchannels prior, during, and after the link was added. All configuration and verification was done via Fabric Manager with confirmation through 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 11 ISL members after removal.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 11 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 11 members.

Step 6 Add a single additional port to the portchannel (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 12 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

- We expect that the single link addition/integration is executed successfully without errors, issues, or problems.
- We expect that the single link addition didn't affect active storage traffic over the portchannel.
- We expect the active storage 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 portchannel group does not disrupt active traffic or services over the channel. Storage traffic was generated to cross the portchannels prior, during, and after the link failure. All configuration and verification was done via 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 12 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 12 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 12 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

- We expect that the single link failure doesn't affect active storage traffic over the portchannel.
- We expect the failure to be detected and reported by the nodes to the management applications/servers.
- We expect the active storage traffic to be load-balanced over the remaining portchannel 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 (shut down) of a single link from an active portchannel group without disrupting active traffic or services over the channel. Storage traffic was generated to cross the portchannels prior, during, and after the link was removed. All configuration and verification was done via 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 12 member ISLs.

Step 3 Generate traffic (with random OXIDs) using the SANtester tool from an edge MDS across a port channel with 12 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 12 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

- We expect that the single link removal (shut down) is executed successfully without errors, issues, or problems.
- We expect that the single link removal didn't affect active storage traffic over the portchannel.
- We expect the active storage traffic to be load-balanced over the remaining portchannel members.
- We expect no CPU or memory problems.

Results

Single Link Remove from Group passed.

Resiliency functionality tests examine whether single component failures cause unexpected storage access disruption. These tests were not performed in Phase 2 for HP due to hardware issues and time constraints.

Resiliency Functionality

The resiliency functionality tests examine whether single component failures cause unexpected storage access disruption.

The following test features were conducted:

- EMC, page 4-91
- NetApp, page 4-95

- HP, page 4-99
- MDS, page 4-102

EMC

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.

The following tests were performed:

- Host Link Failure (Link Pull)—EMC, page 4-91
- Host Link Failure (Port Shutdown)—EMC, page 4-92
- Host Facing Module Failure (OIR)—EMC, page 4-93
- Host Facing Module Failure (Reload)—EMC, page 4-94

Host Link Failure (Link Pull)—EMC

This test verified the Fabric and Host resiliency to a physical link failure when multi-pathing software is running in a host with redundant paths. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. One of the test-host links was pulled to verify traffic re-route to redundant connections. This link was later re-connected to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the Host Link Failure (Link Pull)—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 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

- We expect that traffic is not 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 traffic from non-test-hosts is not affected by the failure or recovery.
- We expect all systems to recover completely from the link failure. Traffic rates and delays return to pre-failure status.
- We expect all the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Host Link Failure (Link Pull)—EMC passed.

Host Link Failure (Port Shutdown)—EMC

This test verified the Fabric and Host resiliency to a port shut down when multi-pathing software is running in a host with redundant paths. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. One of the test-host links was shut down to verify traffic re-route to redundant connections. This link was later re-enabled to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through Fabric Manager .

Test Procedure

The procedure used to perform the Host Link Failure (Port Shutdown)—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 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.

Expected Results

- We expect that traffic is not stopped by the shut down 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 traffic from non-test-hosts is not affected by the failure or recovery.
- We expect all systems to recover completely from the link shut down. Traffic rates and delays return to pre-failure status.

- We expect the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Host Link Failure (Port Shutdown)—EMC passed.

Host Facing Module Failure (OIR)—EMC

This test verified the Fabric and Host resiliency to a host-facing module removal/re-insertion when multi-pathing software is running in a host with redundant paths. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. Test-host facing module was removed to verify traffic re-routing to redundant connections. Edge module is re-inserted to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the Host Facing Module Failure (OIR)—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 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

- We expect that traffic is not 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 traffic from non-test-hosts is not affected by the failure or recovery.
- We expect all systems to recover completely from the module removal/re-insertion. Traffic rates and delays return to pre-failure status.
- We expect all the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Host Facing Module Failure (OIR)—EMC passed.

Host Facing Module Failure (Reload)—EMC

This test verified the Fabric and Host resiliency to a host-facing module reload when multi-pathing software is running in a host with redundant paths. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. Test-host facing module is reloaded to verify traffic re-route to redundant connections. Edge module came back online to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the Host Facing Module Failure (Reload)—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 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

- We expect that traffic is not 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 traffic from non-test-hosts is not affected by the failure or recovery.
- We expect all systems to recover completely from the module reload. Traffic rates and delays return to pre-failure status.
- We expect all the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Host Facing Module Failure (Reload)—EMC 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.

The following tests were performed:

- Host Link Failure (Link Pull)—NETAPP, page 4-95
- Host Link Failure (Port Shutdown)—NETAPP, page 4-96
- Host Facing Module Failure (OIR)—NETAPP, page 4-97
- Host Facing Module Failure (Reload)—NETAPP, page 4-98

Host Link Failure (Link Pull)—NETAPP

This test verified the Fabric and Host resiliency to a physical link failure when multi-pathing software is running in a host with redundant paths. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. One of the test-host links was pulled to verify traffic re-route to redundant connections. This link was later re-connected to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through 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

- We expect that traffic is not 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 traffic from non-test-hosts is not affected by the failure or recovery.

- We expect all systems to recover completely from the link failure. Traffic rates and delays return to pre-failure status.
- We expect all the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- 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 shut down when multi-pathing software is running in a host with redundant paths. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. One of the test-host links was shut down to verify traffic re-route to redundant connections. This link was later re-enabled to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through 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

- We expect that traffic is not stopped by the shut down 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 traffic from non-test-hosts is not affected by the failure or recovery.
- We expect all systems to recover completely from the link shut down. Traffic rates and delays return to pre-failure status.
- We expect the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).

- We expect no CPU or memory problems.

Results

Host Link Failure (Port Shutdown)—NETAPP passed.

Host Facing Module Failure (OIR)—NETAPP

This test verified the Fabric and Host resiliency to a host-facing module removal/re-insertion when multi-pathing software is running in a host with redundant paths. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. Test-host facing module was removed to verify traffic re-routing to redundant connections. Edge module is re-inserted to verify full recovery of failed condition. All configurations and verifications were done via with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the Host Facing 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

- We expect that traffic is not 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 traffic from non-test-hosts is not affected by the failure or recovery.
- We expect all systems to recover completely from the module removal/re-insertion. Traffic rates and delays return to pre-failure status.
- We expect all the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Host Facing Module Failure (OIR)—NETAPP passed.

Host Facing Module Failure (Reload)—NETAPP

This test verified the Fabric and Host resiliency to a host-facing module reload when multi-pathing software is running in a host with redundant paths. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. Test-host facing module is reloaded to verify traffic re-route to redundant connections. Edge module came back online to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through Fabric Manager.

Test Procedure

The procedure used to perform the Host Facing 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

- We expect that traffic is not 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 traffic from non-test-hosts is not affected by the failure or recovery.
- We expect all systems to recover completely from the module reload. Traffic rates and delays return to pre-failure status.
- We expect all the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Host Facing Module Failure (Reload)—NETAPP passed.

HP

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, Veritas DMP for Windows) transparently route all traffic over the remaining link.

The following tests were performed:

- Host Link Failure (Link Pull)—HP, page 4-99
- Host Link Failure (Port Shutdown)—HP, page 4-100
- Host Facing Module Failure (OIR)—HP, page 4-101
- Host Facing Module Failure (Reload)—HP, page 4-101

Host Link Failure (Link Pull)—HP

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 test follows:

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

- 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 passed.

Host Link Failure (Port Shutdown)—HP

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 test follows:

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

- 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 passed.

Host Facing Module Failure (OIR)—HP

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)—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 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

- 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)—HP passed.

Host Facing Module Failure (Reload)—HP

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)—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 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

- 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)—HP passed.

MDS

The MDS resiliency functionality tests induced failures and resets for every major component type in the host-to-storage connectivity fabrics to make sure the fabric design transparently avoided disruption of host and storage traffic. Non-disruptive upgrade from the previous phase version to the current phase version was also tested.

The following tests were performed:

- Active Crossbar Fabric Failover (OIR), page 4-103
- Active Supervisor Failover (OIR), page 4-104

Active Crossbar Fabric Failover (OIR)

This test verified that a removal/re-insertion of an active crossbar-fabric in an edge node causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. An active Crossbar-fabric module was removed to verify the non-disruption of traffic as the other module takes over. Crossbar-fabric module was re-inserted back online to verify full recovery of failed condition. All configurations and verifications were done via 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 an 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

- We expect that traffic is not stopped by the removal or re-insertion of the ACTIVE crossbar-fabric as redundant a module is present.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from the re-insertion.
- We expect the removal and re-insertion to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Active Crossbar Fabric Failover (OIR) passed.

Active Supervisor Failover (OIR)

This test verified that a removal/re-insertion of the active supervisor in an edge node causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. Active Supervisor module was removed to verify the non-disruption of traffic as the standby module takes over. Supervisor module was re-inserted back online (in standby) to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through 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

- We expect that traffic is not stopped by the removal or re-insertion of the ACTIVE supervisor as redundant supervisor modules are present.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from the re-insertion.

- We expect the removal and re-insertion to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- 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 node causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. Active Supervisor module was reloaded to verify the non-disruption of traffic. Supervisor module came back online (in standby) to verify full recovery of failed condition. All configurations were done through Fabric Manager and verifications were done via 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

- We expect that traffic is not stopped by the reload as redundant supervisor modules are present.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from the reload.
- We expect the reload to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- 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 (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. Active Supervisor module is failed over to the standby to verify the non-disruption of traffic. Supervisor module came back online (in standby) to verify full recovery of failed condition. All configurations and verifications were done via 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

- We expect that traffic is not stopped by the fail-over as redundant supervisor modules are present.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from the fail-over.
- We expect the failover and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- 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 (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. The back fan-tray unit was removed to verify the non-disruption of traffic. Fan-tray was re-inserted to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through 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

- We expect that traffic is not stopped by the back fan-tray removal if replaced within the specified time (5 min).
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from the removal/re-insertion.
- We expect the back fan-tray removal/re-insertion to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- 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/re-insertion when portchanneling is running between Edge and Core nodes with multi-module member distribution. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. Core-facing module is removed to verify traffic re-distribution over remaining portchannel members. Core-facing module is re-inserted to verify full recovery of failed condition. All configurations and verifications were done via Fabric Manager with confirmation through CLI.

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 portchannel 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

- We expect that traffic is not stopped by the failure as portchannel 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 is not affected by the failure or recovery.
- We expect all systems to recover completely from the module re-insertion. Traffic rates and delays return to pre-failure status.
- We expect all the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- 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 portchanneling is running between Edge and Core nodes with multi-module member's distribution. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. Core-facing module is reloaded to verify traffic re-distribution over remaining portchannel members. Core-facing module came back online to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through 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 portchannel 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

- We expect that traffic is not stopped by the failure as portchannel 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 is not affected by the failure or recovery.
- We expect all systems to recover completely from the module reload. Traffic rates and delays return to pre-failure status.
- We expect all the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- 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 causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. The front fan-tray unit was removed to verify the non-disruption of traffic. Fan-tray was re-inserted after 5 minute 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 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

- We expect that traffic is not stopped by the front fan-tray removal if not replaced within the specified time (5 min).
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from the removal/re-insertion.
- We expect the front fan-tray removal/re-insertion to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- 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) causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. One of the core nodes was powered-off to verify the non-disruption of traffic. The core node was powered ON to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through 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

- We expect that traffic is not stopped by the complete loss of a core node as a redundant core node is present.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from core node loss/recovery.
- We expect the core node loss to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- 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) causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. One of the core nodes was reloaded to verify the non-disruption of traffic. The core node was online to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through 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

- We expect that traffic is not stopped by the complete loss of a core node as a redundant core node is present.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from core node reload.
- We expect the core node reload to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).

- 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 causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. One of the two power supply units lost power cord connection to verify the non-disruption of traffic. Power supply was plugged back ON to verify full recovery of failed condition. All configurations and verifications were done via CLI with confirmation through 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

- We expect that traffic is not stopped by the power supply loss as a redundant power supply unit is present.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all 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 (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Power Supply Failure (Cord Removal) passed.

Power Supply Failure (Power Off)

This test verified that a loss of a power supply unit in a core node causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. One of the two power supply units was powered-off to verify the non-disruption of traffic. Power supply was powered ON to verify full recovery of failed condition. All configurations and verifications were done via Fabric Manager with confirmation through CLI.

Test Procedure

The procedure used to perform the Power Supply Failure (Power Off) test follows:

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

- We expect that traffic is not stopped by the power supply loss as a redundant power supply unit is present.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from the power loss/recovery.
- We expect the power supply removal/re-insertion to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Power Supply Failure (Power Off) 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

- 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

This test verified that a code upgrade to a core node causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test hosts (Windows and Linux) to the storage array, 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 Fabric Manager with confirmation through CLI.

Test Procedure

The procedure used to perform the SAN OS Code Upgrade test follows:

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 (IOs) from multiple test-hosts (OS: Windows and Linux) to the particular storage array.

Step 3 Generate IO using SAN Tester at line rate to two different line cards.

step3-test_sanos_upgrade.xml

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

- 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 server, etc.)
- We expect no CPU or memory problems.

Results

SAN OS Code Upgrade passed.

Standby Supervisor Failure (OIR)

This test verified that a removal/re-insertion of the STANDBY supervisor in an edge node causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. STANDBY Supervisor module was removed to verify the non-disruption of traffic. Supervisor module was re-inserted and came up online (in standby) to verify full recovery of failed condition. All configurations and verifications were done via 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

- We expect that traffic is not stopped by the removal as the supervisor module is the STANDBY unit.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from the removal/re-insertion.
- We expect the removal/re-insertion to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Standby Supervisor Failure (OIR) passed.

Standby Supervisor Failure (Reload)

This test verified that a reload of the STANDBY supervisor in an edge node causes NO disruption to active services and storage traffic. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. STANDBY Supervisor module was reloaded to verify the non-disruption of traffic. Supervisor module came back online (in standby) to verify full recovery of failed condition. All configurations and verifications were done via 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

- We expect that traffic is not stopped by the reload as supervisor module is the STANDBY unit.
- We expect connectivity from test-hosts to arrays is not affected by the failure or recovery.
- We expect all systems to recover completely from the reload.

- We expect the reload to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Standby Supervisor Failure (Reload) passed.

Unused Module Failure (OIR)

This test verified the Fabric resiliency to an unused module (edge node) reload/removal/re-insertion when there is NO storage traffic through it. Storage traffic (IOs) were generated by the test-hosts (Windows and Linux) to the storage array. An unused module in the Edge node was reloaded/removed/re-inserted to verify that there is NO effect on storage traffic or support services. The module was then re-inserted to verify full recovery of failed condition. All configurations and verifications were done via 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

- We expect that traffic is not stopped by the unrelated failure.
- We expect connectivity from test-hosts to arrays is not affected by the unrelated failure or recovery.
- We expect all systems to recover completely from the module re-insertion.
- We expect all the failure and recovery to be detected and reported by the devices to the management application servers (e.g., Fabric Manager, SYSLOG server, etc.).
- We expect no CPU or memory problems.

Results

Unused Module Failure (OIR) passed.

FCIP Tape Acceleration

The Fiber Channel over IP (FCIP) tape acceleration tests checked both read and write acceleration over varying simulated distances using an ADIC i500 Scalar tape library with IBM LTO3 drives and RedHat Enterprise Linux servers running Veritas NetBackup. Tests also included software and hardware compression.

The following test feature was conducted:

- Tape Read Acceleration, page 4-118
- Tape Write Acceleration, page 4-129

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. The FCIP tape read acceleration tests establish a baseline time for the best possible performance by doing a restore from a tape drive to a host on the same fabric connected by a 4 Gbps fiber channel (not FCIP). The time taken for this restore is the benchmark for later tests using FCIP with and without tape acceleration and with and without software and hardware compression. Media handling times (tape mounting and tape positioning) are not excluded from timing measurements. The simulated distances are 0 km, 100 km and 5000 km. The software used is Veritas NetBackup version 6.0 running on RedHat 4 update 4 Linux servers. The master server is in DCa. This server doubles as a media server. The server in DCb is just a media server. The Shared Storage Option (SSO) is enabled so each server sees both tape drives. The DCa server sees the tape drives locally through fiber channel and the DCb server sees them remotely through FCIP. A representative file system (a copy of the Linux operating system file system with a size of about 8 GB and almost 218,000 files of various types) is restored from tape in each test to either the DCa or the DCb server, depending on the test.

The following tests were performed:

- Tape Read Acceleration—Local Baseline, page 4-118
- Tape Read Acceleration—Remote Baseline, page 4-119
- Tape Read Acceleration—0 km No Compression, page 4-120
- Tape Read Acceleration—100 km No Compression, page 4-121
- Tape Read Acceleration—5000 km No Compression, page 4-122
- Tape Read Acceleration—0 km Hardware Compression, page 4-123
- Tape Read Acceleration—100 km Hardware Compression, page 4-124
- Tape Read Acceleration—5000 km Hardware Compression, page 4-125
- Tape Read Acceleration—0 km Software Compression, page 4-126
- Tape Read Acceleration—100 km Software Compression, page 4-127
- Tape Read Acceleration—5000 km Software Compression, page 4-128

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 a tape drive 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 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 tape drive on the same fabric. Also ensure a restore image of the test data is available and space is available on the host to restore the image. (Test data is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to 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

Tape Read Acceleration—Local Baseline passed.

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 a tape drive to a host on a remote fabric connected by an FCIP using half of an OC-12 (max-bandwidth parameter is 311 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 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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). Compare the local 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

- We expect the restore to 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 throughput to be less than for the local baseline test.
- We expect no CPU or memory problems.

Results

Tape Read Acceleration—Remote Baseline passed.

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 a tape drive to a host on a remote fabric connected by an FCIP using half of an OC-12 (max-bandwidth parameter is 311 Mbps) with no added latency and tape acceleration enabled.

Test Procedure

The procedure used to perform the 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to succeed and the tape read acceleration function will work.
- We expect throughput to be at least as great as the remote baseline test throughput.
- We expect no CPU or memory problems.

Results

Tape Read Acceleration—0 km No Compression passed.

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 a tape drive to 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 enabled.

Test Procedure

The procedure used to perform the 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to succeed and the tape read acceleration function will work.
- We expect throughput to 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

Tape Read Acceleration—100 km No Compression passed.

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 a tape drive to a host on a remote fabric connected by an FCIP using half of an OC-12 (max-bandwidth parameter is 311 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 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 drive 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to succeed and the tape read acceleration function will work.
- We expect throughput to 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

Tape Read Acceleration—5000 km No Compression passed.

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 an FCIP using half of an OC-12 (max-bandwidth parameter is 311 Mbps) with no added latency and with tape acceleration and hardware compression enabled.

Test Procedure

The procedure used to perform the 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to succeed and the tape read acceleration function will work.
- We expect throughput to 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

Tape Read Acceleration—0 km Hardware Compression passed.

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 a tape drive to 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 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to succeed and the tape read acceleration function will work.
- We expect throughput to 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

Tape Read Acceleration—100 km Hardware Compression passed.

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 a tape drive to a host on a remote fabric connected by an FCIP using half of an OC-12 (max-bandwidth parameter is 311 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 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 a restore image of the test data is available and space is available on the host to restore the image. (Test data is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to succeed and the tape read acceleration function will work.

- We expect throughput to 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

Tape Read Acceleration—5000 km Hardware Compression passed.

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 possibly distance by measuring the time for doing a restore from a tape drive to a host on a remote fabric connected by an FCIP using half of an OC-12 (max-bandwidth parameter is 311 Mbps) with no added latency and tape acceleration and software (mode 2) compression enabled.

Test Procedure

The procedure used to perform the 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to succeed and the tape read acceleration function will work.
- We expect throughput to 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

Tape Read Acceleration—0 km Software Compression passed.

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 a tape drive to 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 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 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 restore image of the test data is available and space is available on the host to restore the image. (Test data is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to succeed and the tape read acceleration function will work.
- We expect throughput to 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

Tape Read Acceleration—100 km Software Compression passed.

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 a tape drive to a host on a remote fabric connected by an FCIP using half of an OC-12 (max-bandwidth parameter is 311 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 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 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 a restore image of the test data is available and space is available on the host to restore the image. (Test data is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the restore to succeed and the tape read acceleration function will work.
- We expect throughput to 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

Tape Read Acceleration—5000 km Software Compression 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. The FCIP tape write acceleration tests establish a baseline time for the best possible performance by doing a backup to a tape drive from 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 and with and without software and hardware compression. Media handling times (tape mounting and tape positioning) are not excluded from timing measurements. The simulated distances are 0 km, 100 km and 5000 km. The software used is Veritas NetBackup version 6.0 running on RedHat 4 update 4 Linux servers. The master server is in DCa. This server doubles as a media server. The server in DCb is just a media server. The Shared Storage Option (SSO) is enabled so each server sees both tape drives. The DCa server sees the tape drives locally through fiber channel and the DCb server sees them remotely through FCIP. A representative file system (a copy of the Linux operating system file system with a size of about 8 GB and almost 218,000 files of various types) is backed up to tape in each test from either the DCa or the DCb server, depending on the test.

The following tests were performed:

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 a tape drive from 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 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 tape drive on the same fabric. 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to 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

Tape Write Acceleration—Local Baseline passed.

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 a tape drive from 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 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 drive on different fabrics with no added latency and no 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to 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 throughput may be somewhat less than for the local baseline test.
- We expect no CPU or memory problems.

Results

Tape Write Acceleration—Remote Baseline passed.

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 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 no added latency and tape acceleration enabled.

Test Procedure

The procedure used to perform the 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to succeed and the tape write acceleration function to work.
- We expect throughput to be at least as great as the remote baseline test throughput.
- We expect no CPU or memory problems.

Results

Tape Write Acceleration—0 km No Compression passed.

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 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 enabled.

Test Procedure

The procedure used to perform the 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to succeed and the tape write acceleration function to work.
- We expect throughput to 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

Tape Write Acceleration—100 km No Compression passed.

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 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 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 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 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to succeed and the tape write acceleration function to work.
- We expect throughput to 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

Tape Write Acceleration—5000 km No Compression passed.

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 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 no added latency and tape acceleration and hardware compression enabled.

Test Procedure

The procedure used to perform the 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 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to succeed and the tape write acceleration function to work.
- We expect throughput to 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

Tape Write Acceleration—0 km Hardware Compression passed.

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 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 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 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to succeed and the tape write acceleration function to work.
- We expect throughput to be near the remote baseline test throughput but still less than the local baseline test throughput.
- We expect no CPU or memory problems.

Results

Tape Write Acceleration—100 km Hardware Compression passed.

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 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 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 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 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 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to succeed and the tape write acceleration function to work.
- We expect throughput to 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

Tape Write Acceleration—5000 km Hardware Compression passed.

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 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 no added latency and tape acceleration and software compression enabled.

Test Procedure

The procedure used to perform the 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. 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to succeed and the tape write acceleration function to work.
- We expect throughput to close to the corresponding remote baseline test throughput but still less than the local baseline test.
- We expect no CPU or memory problems.

Results

Tape Write Acceleration—0 km Software Compression passed.

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 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to succeed and the tape write acceleration function to work.
- We expect throughput to 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

Tape Write Acceleration—100 km Software Compression passed.

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 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 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 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 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 is a copy of the root file system of a typical RedHat Enterprise Linux version 4 server with 7,633,836 KB (7.28 GB) of space and 217,000 files and directories.)

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

- We expect the backup to succeed and the tape write acceleration function to work.
- We expect throughput to 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

Tape Write Acceleration—5000 km Software Compression passed.

CHAPTER 5

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 for availability, location and load of content servers. Using the GSS in combination with Cisco's Content Services Switch (CSS) or Cisco's Catalyst 6000 Content Switching Module (CSM) 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 3.0 tests, four GSS's were used in the entire GSS network across both data centers.

Two GSS's were installed at each data center.

The GSS receives DNS queries from client DNS proxies (Local D-Proxies are installed at each branch location in DCAP 3.0 which NS Forwards the dns queries to the GSS's), 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.

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. GSS answers are 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 3.0 configuration, the GSS is authoritative for multiple domain names for which the CSM's 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.

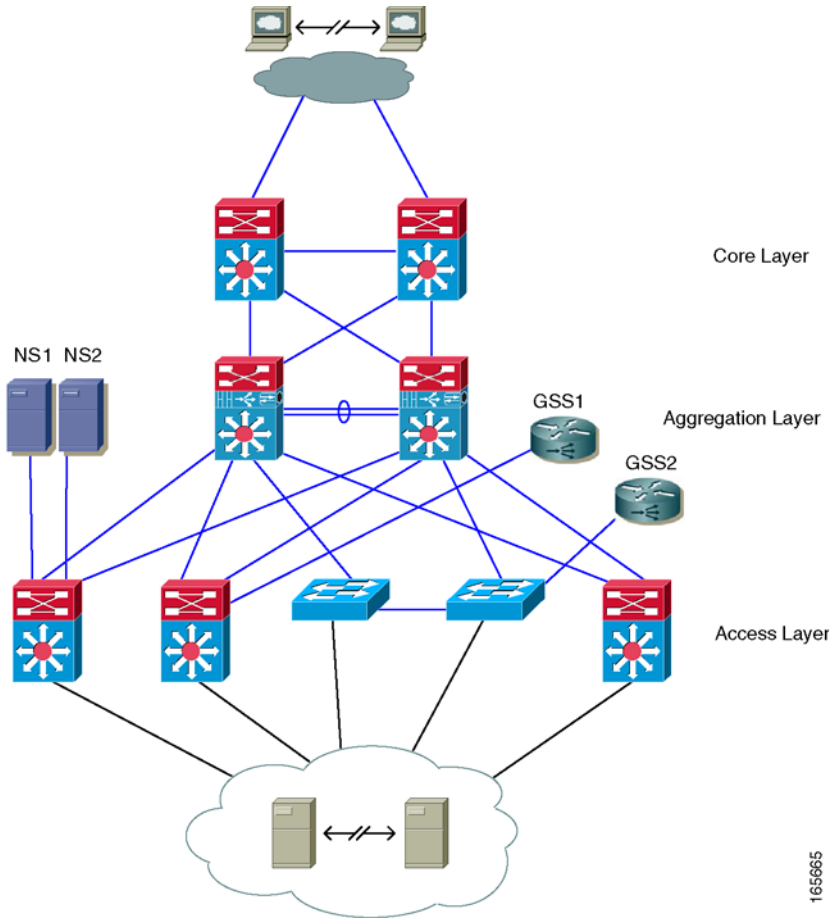
In addition to answering queries directly, the GSS offers the feature of forwarding requests to NS Forwarders, which will return a DNS response packet to the GSS, which in turn returns the exact same response packet to the originally requesting D-Proxy. This can be used for any query type on any domain, and is not limited to the record types supported by the GSS. In DCAP 3.0 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 though the use of CSM's installed at each data center.

GSS Topology

The GSS's are integrated into the existing DCAP 3.0 topology (Figure 5-1) 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 5-1 DCAP GSS Test Topology



Test Results Summary

Table 5-1 summarizes tests executed as part of the Cisco DCAP 3.0 testing initiative. Table 5-1 includes the feature or function tested, the section that describes the feature set the feature or function belongs to, the component tests for each feature or function, and whether the test is new in this phase of DCAP testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested. DCAP is designed to cover critical path areas and augment ongoing regression and systems testing.

Table 5-1 Cisco DCAP 3.0 GSS Testing Summary

Test Suites	Features/Functions	Tests	Results
GSS Test Cases, page 5-4	GSS Test Cases, page 5-4	1. Backup Restore Branch 1 & Branch 3—Complete	CSCsj16464
		2. GSS DNS Processing	
		3. GSS DNS Static Proximity	
		4. Dynamic Proximity (no RESET) Wait Disabled	
		5. Dynamic Proximity (no RESET) Wait Enabled	
		6. Dynamic Proximity (with RESET) Wait Disabled—Complete	
		7. Dynamic Proximity (with RESET) Wait Disabled	
		8. Global Sticky Branch 1 & Branch 3—Complete	
		9. GSS KALAP to CSM using VIP—Complete	
		10. KAL-AP by TAG—Complete	
		11. LB Methods—Complete	CSCsj26410

GSS DDTS Summary

Table 5-2 lists Development Defect Tracking System (DDTS) software bugs with descriptions, and comments filed by the DCAP testing team during Cisco DCAP 3.0 GSS testing. Table 5-3 lists DDTS with descriptions encountered during Cisco DCAP 3.0 WAAS testing.

Table 5-2 Summary of DDTS Filed During Cisco DCAP 3.0 GSS Testing

DDTS	Description
CSCsj16464	sysdb_item_fail error message when clear statistics dns

Table 5-3 Summary of DDTS Encountered During Cisco DCAP 3.0 GSS Testing

DDTS	Description
CSCsj26410	CSM returns load=128 for KAL-AP by VIP when vserver is OOS

GSS Test Cases

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

Functionality critical to global enterprises in Cisco DCAP 3.0 Storage Area Network (SAN) testing is described in the following section. Refer to Cisco Data Center Assurance Program (DCAP) 3.0 Configurations document for test device configurations.

The following tests were performed:

Backup Restore Branch 1 & Branch 3—Complete

This test verified that the GSS database and the GSS network configuration on the primary GSSM, was properly backed up and restored. During the restore process on the primary GSSM, DNS queries were sent to two name servers to verify proper NS forwarding to the active GSS.

Test Procedure

The procedure used to perform the Backup Restore Branch 1 & 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 dca-gss-1.gslb.dcap.com, ensure the GSS status is primary by issuing the **gss status** command; and, on dcb-gss-1.gslb.dcap.com ensure the GSS status is standby by issuing the **gss status** command.

Step 3 On dca-gss-1.gslb.dcap.com create a full backup of your primary GSSM by issuing the **gssm backup full gssm_back** command.

Step 4 On dca-gss-1.gslb.dcap.com make a change to the GSSM database by creating a new source address list called list_1, which includes the source address 1.1.1.1/32.

Step 5 On dca-gss-1.gslb.dcap.com, create another full backup of your primary GSSM by issuing the **gssm backup full gssm_back_new** command.

Step 6 On dca-gss-1.gslb.dcap.com, verify the full backup files were created, and note the size of the files by issuing the **dir** command.

Step 7 On dca-gss-1.gslb.dcap.com stop the GSS by issuing the **gss stop** command.

Step 8 On both client machines, branch1-client-1.cisco.com and branch3-client-1.cisco.com, ensure the primary and secondary name servers are 10.0.10.2 and 10.0.30.2, respectively.

Step 9 While dca-gss-1.gslb.dcap.com is stopped, verify that the clients are still able to resolve DNS via one of the other 3 GSS's. Verify the other 3 GSS's respond with the expected result. Send a DNS A record query from both clients, branch1-client-1.cisco.com and branch3-client-1.cisco.com, to both of the name servers. From branch1-client-1.cisco.com issue the following commands: nslookup set d wwwin-oefin.gslb.dcap.com.

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, force a rotation and deletion of logs by issuing the **rotate-logs** and the **rotate-logs delete-rotated-logs** commands.

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, 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 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 enable real-time logging by issuing the **show log follow** command.

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, verify the DNS global statistics by issuing the **show statistics dns global** command.

Step 14 On dca-gss-1.gslb.dcap.com, restore the full backup file named gssm_back.full by issuing the **gssm restore gssm_back.full** command, and follow the prompts to reboot the GSS.

Step 15 When dca-gss-1.gslb.dcap.com is back online, verify the source address list list_1, which was added into the backup file gssm_back.full, is no longer present in the DB configuration on the GSS.

Step 16 Verify all GSS's respond with the expected result. Send a DNS A record query from both clients, branch1-client-1.cisco.com and branch3-client-1.cisco.com, to both of the name servers. From both clients, issue the following commands: nslookup set d2 wwwin-oefin.gslb.dcap.com

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

- We expect the GSS software to perform a full backup of the GSSM network configuration settings, and the GSSM database, which contains the global server load balancing configuration information.

Results

Backup Restore Branch 1 & Branch 3—Complete passed with exception CSCsj16464.

GSS DNS 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 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 +qr

```
From 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 +qr

```
From 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 +qr

```
From 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 +qr

```
From 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 +qr

```
From 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 +qr

```
From 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 +qr

```
From 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 +qr

```
From 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

- We expect the GSS to respond to various well formed, RFC based DNS queries in the proper manor.

Results

GSS DNS Processing passed.

GSS DNS Static Proximity

This test verified that the GSS responded with the correct answer(s) based on the source address of the d-proxy.

Test Procedure

The procedure used to perform the GSS DNS Static Proximity test follows:

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

- We expect the GSS to respond with the correct answer(s) based on the source address of the d-proxy.

Results

GSS DNS Static Proximity passed.

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 datacenter to each datacenter'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

- 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 datacenter to each datacenter'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

- 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 datacenter.

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

- We expect that the GSS will properly service dns requests for a dns rule that is enabled for dynamic proximity and Wait disabled while the name server being probed issue sa TCP RST in response to the DRP Probe. The next clause in the dns rule should be matched.

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 datacenter.

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

- We expect that the GSS will properly service dns requests for a dns rule that is enabled for dynamic proximity and Wait Enabled while the name server being probed issues a TCP RST. The next clause in the dns rule should be matched.

Results

Dynamic Proximity (with RESET) Wait Disabled passed.

Global Sticky Branch 1 & 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 & 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

- 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 & Branch 3—Complete passed.

GSS KALAP to CSM using VIP—Complete

This test verified GSS kalap to CSM using TAG.

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

- We expect no CPU or memory problems.

Results

GSS KALAP to CSM using VIP—Complete passed.

KAL-AP by TAG—Complete

This test verified 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

- We expect no CPU or memory problems.

Results

KAL-AP by TAG—Complete passed with exception CSCsj26410.

LB Methods—Complete

This test verified GSS kalap to CSM using TAG.

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 +qr

```
From 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

- 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 6

Wide Area Application Services (WAAS)

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).

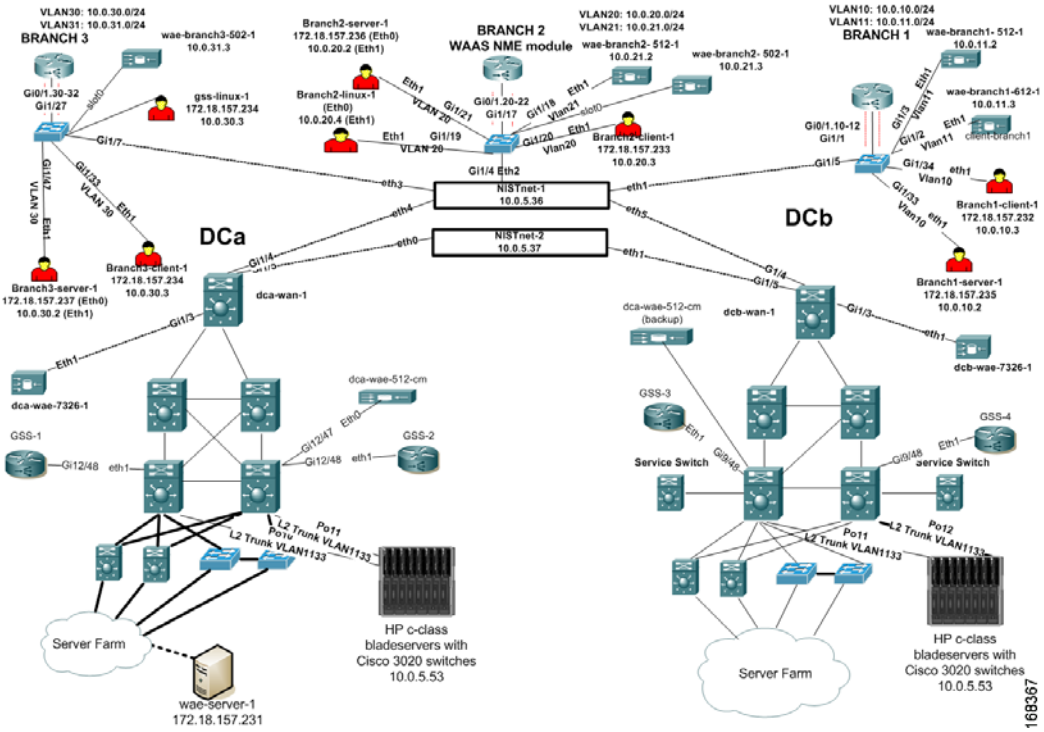
The solution provides the LAN-like performance across the WAN through a combination of technologies, including:

- 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 or Policy-Based Routing (PBR) to intercept traffic and transparently forward it to the local Cisco WAE on both sides of the network (Figure 6-1).

Figure 6-1 DCAP WAAS Test Topology



WAAS Test Results Summary

Table 6-1 summarizes tests executed as part of the Cisco DCAP 3.0 testing initiative. Table 6-1 includes the feature or function tested, the section that describes the feature set the feature or function belongs to, the component tests for each feature or function, and whether the test is new in this phase of DCAP testing.

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. In Table 6-1, where applicable, pointers to relevant portions of these documents are provided for reference purposes.

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

Note Test results are unique to technologies covered and actual scenarios in which they were tested. DCAP is designed to cover critical path areas and augment ongoing regression and systems testing.

Table 6-1 Cisco DCAP 3.0 WAAS Testing Summary

Test Suites	Features/Functions	Tests	Results
Baseline, page 6-6	Upgrades, page 6-7 SRND Page 19: Central Manager Maintenance: 14-1: Upgrading	1. Central Manager CLI Upgrade WAE512 (Standby)	
		2. Central Manager GUI Upgrade WAE512 (Primary)	
		3. Edge CLI Upgrade WAE612	CSCsi49779
		4. Core CLI Upgrade WAE7326	CSCsi49779
		5. Core GUI Upgrade WAE7326	CSCsi49779
		6. Edge CLI Upgrade WAE502	CSCsi49779 CSCsi69388
		7. Edge GUI Upgrade WAE502	CSCsi49779 CSCsi69388
		8. Edge GUI Upgrade WAE512	CSCsi49779
	Device Management, page 6-14	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	
	Reliability, page 6-18 SRND Page 19: Central Manager Maintenance: 14-24: Rebooting a Device	1. Central Manager reload WAE512	
		2. Edge Reload WAE502	CSCsi69388 CSCsi75538
		3. Edge Reload WAE512	CSCsi75538
		4. Core Reload WAE7326	CSCsi75538
	Redundancy, page 6-21 SRND: Page 19: Central Manager SRND: Page 23: Hashing SRND: Page 23: Masking SRND: Page 39: Standby Interface Maintenance: 14-20: CM Failover	1. Active Central Manager failure	CSCsi93903 CSCsh97770
		2. Active Interface Failure and Recovery with Hash Assign	
		3. Active Interface Failure and Recovery with Mask Assign	CSCsi05906
	WCCP, page 6-26 SRND: Page 20: Interception Methods SRND: Page 23: Hashing	1. WCCPv2 Basic Configuration on Edge 2811	
		2. WCCPv2 Basic Configuration on Edge 2821	
		3. WCCPv2 Functionality on Core WAE7326	
		4. WCCPv2 Functionality on Edge WAE 512	
		5. WCCPv2 Functionality on Edge 3845	
		6. WCCPv2 Functionality on Core Sup720	
	NTP, page 6-33	1. NTP Functionality	

Table 6-1 Cisco DCAP 3.0 WAAS Testing Summary (continued)

Test Suites	Features/Functions	Tests	Results
Optimization (DRE/TFO/LZ), page 6-35	Acceleration, page 6-35	1. FTP Acceleration Branch 1	
		2. FTP Acceleration Branch 2	
		3. FTP Acceleration Branch 3	
		4. HTTP Acceleration Branch 1	
		5. HTTP Acceleration Branch 2	CSCsh92758
		6. HTTP Acceleration Branch 3	
	CIFS/WAFS Performance, page 6-43 SRND: Page 20: CIFS Compatibility	1. WAFS Configuration Verification	
		2. CIFS Cache Hit Benchmark Branch 1	CSCsi58809
		3. CIFS Cache Hit Benchmark Branch 2	CSCsi58809
		4. CIFS Cache Hit Benchmark Branch 3	CSCsi58809
		5. CIFS Cache Miss Benchmark Branch 1	CSCsi58809
		6. CIFS Cache Miss Benchmark Branch 2	CSCsi58809
		7. CIFS Cache Miss Benchmark Branch 3	CSCsi58809
		8. CIFS Native WAN Benchmark Branch 1	
		9. CIFS Native WAN Benchmark Branch 2	
		10. CIFS Native WAN Benchmark Branch 3	
		11. CIFS Verification WAE502	CSCsi58809
		12. CIFS Verification WAE512	CSCsi58809
		13. CIFS Verification WAE612	CSCsi58809

WAAS DDTS Summary

Table 6-2 lists Development Defect Tracking System (DDTS) software bugs with descriptions, and comments filed by the DCAP testing team during Cisco DCAP 3.0 WAAS testing. Table 6-3 lists DDTS with descriptions encountered during Cisco DCAP 3.0 WAAS testing. Table 6-4 lists DDTS with descriptions of interest not encountered during Cisco DCAP 3.0 WAAS testing.

Table 6-2 Summary of DDTS Filed During Cisco DCAP 3.0 WAAS Testing

DDTS	Description
CSCsi69388	WAE-502: Routing of statically defined routes after reload incorrect. Statically defined routes are automatically sent out the primary interface. This is only seen on the WAE network module. Workaround: Reconfigure static route statement.
CSCsi75538	WAE-502: Startup and running config different after copy run start

Table 6-3 Summary of DDTS Encountered During Cisco DCAP 3.0 WAAS Testing

DDTS	Description
CSCsi44512	WAFS Admin log is flooded with SMB signing messages, not manageable
CSCsi05906	(6k) WCCP:appliance failover does not update TCAM adjacency. When a standby interface on a WAE takes over w/ MASK assigned redirection traffic is blackholed. Workaround: restart WCCP
CSCsh92758	WAAS exiting flows are not purged on the WAE when link goes down. Overload mode possible to reach since flows are not purged.
CSCsi49779	EPM Enabled After Upgrade from 4.0.7 to 4.0.9
CSCsi93903	nodemgr timestamp in syslog incorrect
CSCsh97770	Show cms info should show backup CM when failed over
CSCsh85073	discrepancy in time when device goes from online to offline state

Table 6-4 Summary of DDTS of Interest Not Encountered During Cisco DCAP 3.0 WAAS Testing

DDTS	Description
CSCsi58809	Synq deadlock hang. Conditions: Client infected with a virus that blasts the network, sending SYN packets to blocks of addresses on port 445. The problem can also be caused by flooding SYN packets on port 139. Workaround: Disable CIFS Auto-discovery on edge devices
CSCsi66928	Need means to accommodate with virus/scanner attacks on CIFS AD ports
CSCsi44131	Wrong WAFS core selected with CIFS AD
CSCsg11506	EPM breaks connections if it doesn't intercept both traffic directions
CSCsi48683	CMS deadlock when preinter driver update is received
CSCsj00523	HTTP fail when WCCP uses GRE for both redirection and packet return. Use L2 redirect when connecting to a 6500.
CSCsh92695	(6k) High CPU on the Cat6K Sup720 with WCCP GRE redirection. Use L2 redirect when connecting to a 6500.
CSCsi28118	FTP behavior of Central manager appears to be non-RFC compliant
CSCsi65531	Exclude CIFS Requests from Connected Cores from CIFS AD
CSCsi88461	WAAS ip access-list counter for snmp-server doesn't work
CSCsi46049	Kernel KDB command is on by default in CM GUI
CSCsi78386	TACACS 'Security Word' Validation Needed for CM

Table 6-4 Summary of DDTS of Interest Not Encountered During Cisco DCAP 3.0 WAAS Testing

DDTS	Description
CSCsi10402	Portchannel :Unable to ping default gateway if 1st interface is shutdown
CSCsh98343	WCCP redirect-list and mask-acl merge results in wrong redirect info. **Workaround:** Configure a separate WCCP redirect-list ACL for each WCCP service (61 and 62) The ACL associated with a service should use the 'any' keyword in the position that matches the service group mask. For example, when applying a redirect-list to service group 61, the access-list should have the following format: access-list 100 permit ip any <network> <inverse_mask> Note that the 'any' keyword is used in the source position of the ACL. Since service group 61 masks on the source IP address, there is no potential conflict between the bits used by the mask-acl and the bits used by the WCCP redirect-list ACL.

WAAS Test Cases

Functionality critical to global enterprises in Cisco DCAP 3.0 Wide Area Application Services (WAAS) testing is described in the following sections. Refer to Cisco Data Center Assurance Program (DCAP) 3.0 Configurations document for test device configurations.

- Baseline, page 6-6
- Optimization (DRE/TFO/LZ), page 6-35

Baseline

The suite of baseline tests was created to verify the basic configuration and functionality of several integral features necessary for the WAAS software to perform correctly. The functionality of the Simple Network Management Protocol (SNMP) on Wide-Area Application Engine (WAE) devices serving as the edge, core, and central manger device was first tested. The reliability of the WAE devices configurations after reload was also tested. The Web Cache Communication Protocol version 2 (WCCPv2), the method used for TCP-based traffic interception and redirection, was then tested on the core and edge WAE devices, as well as, on the core and branch routers. The next protocol tested was the Network Time Protocol (NTP) which plays an important role in the WAAS solution. Each WAE devices clocks must be synchronized in order for the WAAS software to work properly. The NTP protocols functionality was verified on core, edge and Central Manager WAE devices.

The following test features were conducted:

- Upgrades, page 6-7
- Device Management, page 6-14
- Reliability, page 6-18
- Redundancy, page 6-21
- WCCP, page 6-26
- NTP, page 6-33

Upgrades

This test verifies that the Cisco IOS upgrade process works correctly.

The following tests were performed:

Central Manager CLI Upgrade WAE512 (Standby)

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 central manger 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 Central Manager CLI Upgrade WAE512 (Standby) test follows:

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-wae-512-cm 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 After downloading the new version of software to a FTP server issue the **copy ftp install 172.18.177.132 / tftpboot/WAAS-4.0.9.10-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.

Continue monitoring the console until you have verified the new image has booted properly and that you can once again log into the device.

Step 8 Once the WAE has rebooted issue the **show version** command to verify it 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 before the upgrade.

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

- We expect the Central Manager WAE to upgrade without error.
- We expect the Central Manager WAE to boot the new image without error.
- We expect the configuration to be the same after the upgrade.

Results

Central Manager CLI Upgrade WAE512 (Standby) passed.

Central Manager GUI Upgrade WAE512 (Primary)

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 central manger mode to be upgraded to WAAS version under test via the GUI. The configuration was verified before and after the upgrade to verify that no discrepancies were seen.

Test Procedure

The procedure used to perform the Central Manager GUI Upgrade WAE512 (Primary) test follows:

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-wae-512-cm 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 -> dca-wae-512-cm -> 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 Begin monitoring the device via the console and then from the central manger GUI navigate as follows:

Devices -> dca-wae-512-cm -> 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

- We expect the Central Manager WAE to upgrade without error.
- We expect the configuration to be the same after the upgrade.

Results

Central Manager GUI Upgrade WAE512 (Primary) passed.

Edge 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 Edge 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 172.18.177.132 / tftpboot/WAAS-4.0.9.10-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 it 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

- We expect the edge application-accelerator WAE to upgrade without error.
- We expect the configuration to be the same after the upgrade.

Results

Edge CLI Upgrade WAE612 passed with exception CSCsi49779.

Core CLI Upgrade WAE7326

The WAE devices can be upgraded via the Central Mangager GUI or via the device CLI. This test verified the ability of a WAE-7326 running in central manger mode to be upgraded to WAAS version under test via the CLI. The configuration was verified before and after the upgrade to verify that no discrepancies were seen.

Test Procedure

The procedure used to perform the Core CLI Upgrade 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 On dca-wae-7326-1 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 172.18.177.132 / tftpboot/WAAS-4.0.9.10-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 it 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

- We expect the core WAE to upgrade without error.
- We expect the configuration to be the same after the upgrade.

Results

Core CLI Upgrade WAE7326 passed with exception CSCsi49779.

Core GUI Upgrade WAE7326

The WAE devices can be upgraded via the Central Mangager GUI or via the device CLI. This test verified the ability of a WAE-7326 running in core application-accelerator mode to be upgraded to WAAS version under test via the GUI 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 GUI Upgrade 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 On dcb-wae-7326-1 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

- We expect the edge application-accelerator WAE to upgrade without error.
- We expect the configuration to be the same after the upgrade.

Results

Core GUI Upgrade WAE7326 passed with exception CSCsi49779.

Edge CLI Upgrade WAE502

The WAE devices can be upgraded via the Central Mangager GUI or via the device CLI. This test verified the ability of a WAE-502 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 CLI Upgrade 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 On wae-branch2-502-1 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 172.18.177.132 / tftpboot/WAAS-4.0.9.10-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 Issue the **copy running-config startup-config** and **reload** commands to first save the configuration and then reload the WAE.

The WAE-502 has no console so monitoring the reload will not be an option.

Step 8 Once the WAE has rebooted issue the **show version** command to verify it 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

- We expect the edge application-accelerator WAE to upgrade without error.
- We expect the configuration to be the same after the upgrade.

Results

Edge CLI Upgrade WAE502 failed CSCsi49779, CSCsi69388.

Edge GUI Upgrade WAE502

The WAE devices can be upgraded via the Central Mangager GUI or via the device CLI. This test verified the ability of a WAE-502 running in edge application-accelerator mode to be upgraded to WAAS version under test via the GUI 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 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 On wae-branch3-502-1 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-branch3-502-1 -> Update Software

Click the submit button to initialize the upgrade.

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

- We expect the edge application-accelerator WAE to upgrade without error.
- We expect the configuration to be the same after the upgrade.

Results

Edge GUI Upgrade WAE502 failed CSCsi49779, CSCsi69388.

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 GUI 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

- We expect the edge application-accelerator WAE to upgrade without error.
- We expect the configuration to be the same after the upgrade.

Results

Edge GUI Upgrade WAE512 passed with exception CSCsi49779.

Device Management

The Simple Network Management Protocol (SNMP) is an application layer protocol that facilitates the exchange of management information between network devices. It is part of the Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite. SNMP enables network administrators to manage network performance, find and solve network problems, and plan for network growth. The tests performed verified the functionality and resiliency of the device as a series of SNMP Walks were performed on the WAE devices.

The following tests were performed:

- SNMP Central Manager MIB Walk-WAE512, page 6-15
- SNMP Core MIB Walk-WAE7326, page 6-15

- SNMP Edge MIB Walk-WAE502, page 6-16
- SNMP Edge MIB Walk-WAE512, page 6-16
- SNMP Edge MIB Walk-WAE612, page 6-17

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's 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

- We expect all SNMP walks to run without error.
- We expect no tracebacks or crashes to occur on the DUT.

Results

SNMP Central Manager MIB Walk-WAE512 passed.

SNMP Core MIB Walk-WAE7326

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's of the MIB tree of a WAE-7326 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 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 **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

- We expect all SNMP walks to run without error.
- We expect no tracebacks or crashes 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's 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

- We expect all SNMP walks to run without error.
- We expect no tracebacks or crashes to occur on the DUT.

Results

SNMP Edge MIB Walk-WAE502 passed.

SNMP Edge 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's of the MIB tree of a WAE-512 device did not cause any memory 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-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

- We expect all SNMP walks to run without error.
- We expect no tracebacks or crashes to occur on the DUT.

Results

SNMP Edge MIB Walk-WAE512 passed.

SNMP Edge MIB Walk-WAE612

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's of the MIB tree of a WAE-612 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-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

- We expect all SNMP walks to run without error.
- We expect no tracebacks or crashes to occur on the DUT.

Results

SNMP Edge MIB Walk-WAE612 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.

The following tests were performed:

- Central Manager reload WAE512, page 6-18
- Edge Reload WAE502, page 6-19
- Edge Reload WAE512, page 6-20
- Core Reload WAE7326, page 6-21

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

- 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 device GUI to be accessible and that all devices will show up as online.

Results

Central Manager reload WAE512 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 Isolate the WAE so that it is the only active WAE at the branch.

Step 9 From Branch 2 initiate an FTP file transfer from an FTP server in DCB. Verify the file is opimized with the **show tfo connection server-ip 201.1.33.12**, **show statistics tfo**, **show statistics tfo savings**, and **show statistics dre** commands.

Step 10 Bring back up the WAE that was brought down.

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

- 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 WAE502 failed CSCsi69388, 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

- 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 with exception CSCsi75538.

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.

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

- We expect the WAE device to come online after the reload.
- We expect that the configuration to be the same after the reload.
- We expect the WAE to accelerate traffic after the reload.

Results

Core Reload WAE7326 passed with exception CSCsi75538.

Redundancy

Redundancy is provided within WAAS a number of ways. The WAE devices themselves can be clustered into a farm with WCCP and in the event of a WAE failure, WCCP will reconfigure and redistribute connections among the remaing WAE's in the farm. At the device level, each WAE can be deployed with

redundant links into the network in and active/standby fashion. Management of the WAAS network that is handled by the Central Manager can also be deployed redudnantly with a second WAE device running in Central Manager mode.

The following tests were performed:

- Active Central Manager failure, page 6-22
- Active Interface Failure and Recovery with Hash Assign, page 6-23
- Active Interface Failure and Recovery with Mask Assign, page 6-25

Active Central Manager failure

The WAAS Central Manager is the main configuration tool for WAAS network. As expected, WAAS continues to function even when the primary Central Manager is offline. However, no configuration changes can be made. Promotion of the standby Central Manager to the primary Central Manager is configured from the CLI of the standby Central Manager by changing the Central Manager role from standby to primary. When the failed primary Central Manager is back online, it must be configured for a standby CM role.

This test verified the standby Central Manager was able to be promoted to an active role. Once active it was verified that the WAE devices registered correctly. The ability for the initial active (new standby) Central Manager to become active was then verified. Finally the demotion of the old standby was verified and the network was validated that it was in proper working order.

Test Procedure

The procedure used to perform the Active Central Manager 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 the status of the primary and backup Central Managers by issuing the **show device-mode current** and **show cms info** commands.

Step 3 Verify the ability to log into the active central-manager via the HTTP GUI and verify the inablity to log into the standby Central Manager.

Step 4 Issue the **show running-config** on the active and standby Central Managers.

Step 5 Simulate a Central Manager failure on the active manager by shutting down the inband interface to DCA.

Step 6 Log into the standby Central Manager in DCB via the CLI and promote it to the primary role by issuing the **central-manager role primary** command.

Step 7 Verify the ability to log into the new active central-manager via the HTTP GUI.

Step 8 There will be a brief period of time after the promotion of the standby when the devices are not yet registered with the new active Central Manager. During this time verify traffic is accelerated as expected.

Initiate an FTP transfer from branch3 to DCB and verify acceleration occured on the WAE at branch 3 by issuing the **show tfo connection summary** command. Verify that the WAE sees the new active Central Manager as the primary by issuing the **show cms info** command.

Step 9 Verify that all devices except for the new standby Central Manager come online in the GUI.

Step 10 Verify the only changes to the new active Central Managers configuration is the **central-manager role** and **central-manager address** config.

Step 11 Bring up the link on the initial primary Central Manager and issue the **central-manager role primary** command.

Without demoting one of the two managers there will be 2 primary Central Managers configured. Verify that both go to standby mode in this situation.

Step 12 Demote the Central Manager in DCB to a standby role with the **central-manager role standby** and **central-manager address 101.1.33.4** commands. Re-enable central management on the new primary manager with the **cms enable** command.

Step 13 Verify in the Central Manager GUI that the primary and standby Central Managers re-assume the appropriate roles and that all devices in the network register.

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

- We expect the standby CM to take over when the active fails once it has been manually promoted via the CLI.
- We expect the necessary configuration to be synchronized on the primary and standby Central Managers.
- We expect the primary Central Manager to take back over when it is back online when it is promoted to Central Manager via the CLI.
- We expect both Central Managers to go to standby mode when they are both configured as primary.
- We expect the standby Central Manager to go back to standby when it is demoted via the CLI.

Results

Active Central Manager failure passed with exception CSCsi93903, CSCsh97770.

Active Interface Failure and Recovery with Hash Assign

The standby interface is configured to protect from network interface, link, and switch failure. The default priority of the physical interface is 100. GigabitEthernet 2/0 is configured with a higher priority of 105. It is the active interface with the IP address when the WAE is online.

This test verified the standby interface becomes active when the primary active interface goes down and that traffic passed as expected. The WCCP redirect method used is L2-redirect with Hash Assignment.

Test Procedure

The procedure used to perform the Active Interface Failure and Recovery with Hash Assign test follows:

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 with the **show running-config** command that the WAE is configured with an active/standby network configuration on GigabitEthernet 1/0 and GigabitEthernet 2/0.

The primary interface, GigabitEthernet 2/0, should be configured with a priority of 105. The standby interface, GigabitEthernet 1/0, should be configured with the default priority. The default priority for the standby in this configuration is 100.

Step 3 On dcb-wan-1 verify the status of the ports connecting to the active and standby interfaces with the **show interface *mod/port*** status and **show spanning-tree interface *mod/port*** commands.

Step 4 Verify the method of WCCP assignment is the default HASH assign on the WAE with the **show running-config** and **show wccp services detail** commands. If necessary configure the device to use hash, not mask, assign by issuing the **no wccp tcp-promiscuous router-list-num 1 l2-redirect mask-assign assign-method-strict** and **wccp tcp-promiscuous router-list-num 1 l2-redirect** commands.

Step 5 Verify the layer 2 information ARP and CEF data on the WCCP router. On the WAE issue the **show standby** and **show interface** commands. On the WCCP router issue the **show ip arp *standby ip*** and **show ip cef *standby ip*** commands.

Verify that the ARP and CEF info on the router and WAE agree. Each should refer to the correct MAC address of the active interface on the WAE.

Step 6 Begin sending pings in 1 second intervals from a server in DCB to the WAE.

Step 7 Fail the active interface on the WAE with the **shutdown** command.

Step 8 Verify the standby port takes over with the **show standby** and that pings to the device are once again successful. Stop the pings and calculate the approximate downtime. One lost ping is equivalent to 1s of downtime.

Step 9 Verify the layer 2 information ARP and CEF data on the WCCP router. On the WAE issue the **show interface** command. On the WCCP router issue the **show ip arp *standby ip*** and **show ip cef *standby ip*** commands.

Verify that the ARP and DEF info on the router and WAE agree. Each should refer to the correct MAC address of the new active interface on the WAE.

Step 10 Verify connectivity is not disrupted by initiating an FTP file transfer to a server in DCB from a remote branch client.

Step 11 Bring back up the interface on the WAE that was shutdown with the **no shutdown** command. Flap the standby interface to get it back to the default setup. Verify the standby group is back to normal by issuing the **show standby** command.

Step 12 If MASK assign was unconfigured previously reconfigure it on dcb-wae-7326-1 with the **wccp tcp-promiscuous router-list-num 1 l2-redirect mask-assign assign-method-strict** 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

- We expect the standby interface to become active when the primary is failed.
- We expect traffic to pass normally after the failure.

Results

Active Interface Failure and Recovery with Hash Assign passed.

Active Interface Failure and Recovery with Mask Assign

The standby interface is configured to protect from network interface, link, and switch failure. The default priority of the physical interface is 100. GigabitEthernet 2/0 is configured with a higher priority of 105. It is the active interface with the IP address when the WAE is online.

This test verified the standby interface becomes active when the primary active interface goes down. After the failure it is verified that the primary takes back over when network connectivity is re-established.

Test Procedure

The procedure used to perform the Active Interface Failure and Recovery with Mask Assign test follows:

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 with the **show running-config** command that the WAE is configured with an active/standby network configuration on GigabitEthernet 1/0 and GigabitEthernet 2/0.

The primary interface, GigabitEthernet 2/0, should be configured with the default priority of 105. The standby interface, GigabitEthernet 1/0, should be configured with a higher priority than the primary interface. The default priority for the standby in this configuration is 100.

Step 3 On dcb-wan-1 verify the status of the ports connecting to the active and standby interfaces with the **show interface *mod/port*** status and **show spanning-tree interface *mod/port*** commands.

Step 4 Verify the method of WCCP assignment is mask-assign on the WAE with the **show running-config** and **show wccp services detail** commands.

Step 5 Verify the layer 2 information ARP, CEF, and adjacency data on the WCCP router. Verify that the active interface's MAC address is known in the tcam table on the SP of the WCCP router. On the WAE issue the **show standby** and **show interface** commands. On the WCCP router issue the **show ip arp *standby ip***, **show ip cef *standby ip*** and **remote command switch show tcam adj** commands.

Step 6 Begin sending pings in 1 second intervals from a server in DCB to the WAE.

Step 7 Fail the active interface on the WAE with the **shutdown** command.

Step 8 Verify the standby port takes over with the **show standby** and that pings to the device are once again successful. Stop the pings and calculate the approximate downtime. One lost ping is equivalent to 1s of downtime.

Step 9 Verify the layer 2 information ARP, CEF, and adjacency data on the WCCP router. Verify that the active interface's MAC address is known in the tcam table on the SP of the WCCP router. On the WAE issue the **show interface** command. On the WCCP router issue the **show ip arp *standby ip***, **show ip cef *standby ip*** and **remote command switch show tcam adj** commands.

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

- We expect the standby interface to become active when the primary is failed.
- We expect traffic to pass normally after the failure.

- We expect that the primary interface will take back over and pass traffic when network connectivity is re-established.

Results

Active Interface Failure and Recovery with Mask Assign failed CSCsi05906.

WCCP

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.

The following tests were performed:

- WCCPv2 Basic Configuration on Edge 2811, page 6-26
- WCCPv2 Basic Configuration on Edge 2821, page 6-27
- WCCPv2 Functionality on Core WAE7326, page 6-29
- WCCPv2 Functionality on Edge WAE 512, page 6-30
- WCCPv2 Functionality on Edge 3845, page 6-30
- WCCPv2 Functionality on Core Sup720, page 6-32

WCCPv2 Basic Configuration on Edge 2811

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 2811 ISR.

Test Procedure

The procedure used to perform the WCCPv2 Basic Configuration on Edge 2811 test follows:

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 FastEthernet0/1.32** 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 FastEthernet0/1.30** 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, 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

- 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 Basic Configuration on Edge 2811 passed.

WCCPv2 Basic Configuration on Edge 2821

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 2821 test follows:

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 GigabitEthernet0/1.22** 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 GigabitEthernet0/1.20** 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 GigabitEthernet0/1.21** 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, 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

- 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 Basic Configuration on Edge 2821 passed.

WCCPv2 Functionality on Core WAE7326

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.

This test verified the basic configuration and functionality of WCCPv2 on a Core WAE device. The Core device is a WAE-7326.

Test Procedure

The procedure used to perform the WCCPv2 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 device(s) 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**.

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

- 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 Functionality on Core WAE7326 passed.

WCCPv2 Functionality on Edge WAE 512

The Web Cache Communication Protocol (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.

This test verified the basic configuration and functionality of WCCPv2 on a Edge WAE device. In this case the device is a WAE-512.

Test Procedure

The procedure used to perform the WCCPv2 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 device(s) 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

- 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 Functionality on Edge WAE 512 passed.

WCCPv2 Functionality on Edge 3845

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 Functionality on Edge 3845 test follows:

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 GigabitEthernet0/1.12** 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 GigabitEthernet0/1.10** 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 GigabitEthernet0/1.11** 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 1.

Note If no Loobpack address is configured on the router, 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 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

- 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 Functionality on Edge 3845 passed.

WCCPv2 Functionality on Core Sup720

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 Supervisor 720. 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.

Test Procedure

The procedure used to perform the WCCPv2 Functionality on Edge 3845 test follows:

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 port(s) from the WAN 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 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 configuered 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** command.

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 1.

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

- 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 Functionality on Edge 3845 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.

The following test was performed:

- NTP Functionality

NTP 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. This is because timestamps 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 device(s). Through the Central Manager GUI, The Central Manager was first synchronized to a NTP server that is used throughout our 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 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:* 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 3 Verify the NTP configuration on the active and standby Central Manger by issuing the **show running-config** command via the CLI.

Step 4 Verify the CM clock is in sync with the global NTP server which is located on goel(172.18.177.132). SSH to goel and issue **ps -ef | grep /usr/lib/inet/xntpd** to verify the NTP daemon is running. The issue the **date** command at the prompt to get the time. On the CM issue the **show clock** command and verify that the two timestamps are in sync.

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."

Hover the mouse pointer over the icons and click the last one that should have a pop up box that reads *Force Settings on all Devices in Group*. You will see a pop up 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, 201.1.33.4).

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

- We expect the Central Manger 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 Manger clock.

Results

NTP Functionality passed.

Optimization (DRE/TFO/LZ)

The basic configuration for Wide Area File Services (WAFS) implementation was first verified in this suite. The Common Internet File System (CIFS) functionality through the WAE devices was then verified. Finally, the WAFS Benchmark tool was used to baseline the open, save, and close times for several different sizes ranging from 50k to 2MB of Microsoft Office files from a centralized file share on a Windows 2003 file server across the WAN with, and without, optimization.

The following test features were conducted:

- Acceleration, page 6-35
- CIFS/WAFS Performance, page 6-43

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.

The following tests were performed:

- FTP Acceleration Branch 1
- FTP Acceleration Branch 2
- FTP Acceleration Branch 3
- HTTP Acceleration Branch 1
- HTTP Acceleration Branch 2
- HTTP Acceleration Branch 3

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 randomly sized file between 500Kb and 100Mb. 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 1 had 6ms 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 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 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 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

- We expect FTP traffic to pass when WAAS is enabled.
- We expect FTP GET response times to decrease with WAAS acceleration enabled.

Results

FTP Acceleration Branch 1 passed.

FTP Acceleration Branch 2

This test verified the ability of WAAS to accelerate FTP traffic. From Branch 2, 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 17ms latency and T1 bandwidth to the file server in the data center.

Test Procedure

The procedure used to perform the FTP Acceleration Branch 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 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 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 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 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

- We expect FTP traffic to pass when WAAS is enabled.
- We expect FTP GET response times to decrease with WAAS acceleration enabled.

Results

FTP Acceleration Branch 2 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 3 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 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 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 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

- We expect FTP traffic to pass when WAAS is enabled.
- We expect FTP GET response times to 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, 10 simulated clients making requests of 50-100k 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 Shenick 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 in the 50k-100k 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 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 Shenick 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 in the 50k-100k 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 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 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

- We expect HTTP traffic to pass when WAAS is enabled.
- We expect HTTP GET response times to decrease with WAAS acceleration enabled.

Results

HTTP Acceleration Branch 1 passed.

HTTP Acceleration Branch 2

This test verified the ability of WAAS to accelerate HTTP traffic. From Branch 2, 10 simulated clients making requests of 50-100k 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 2 had 17ms latency and T1 bandwidth to the file server in the data center.

Test Procedure

The procedure used to perform the HTTP Acceleration Branch 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 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 Shenick 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 in the 50k-100k 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 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 Shenick 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 in the 50k-100k 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 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 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

- We expect HTTP traffic to pass when WAAS is enabled.
- We expect HTTP GET response times to decrease with WAAS acceleration enabled.

Results

HTTP Acceleration Branch 2 passed with exception CSCsh92758.

HTTP Acceleration Branch 3

This test verified the ability of WAAS to accelerate HTTP traffic. From Branch 3, 10 simulated clients making requests of 50-100k 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. Branch 3 had 70ms latency and T1 bandwidth to the file server in the data center.

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 Shenick 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 in the 50k-100k 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 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 Shenick 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 in the 50k-100k 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 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 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

- We expect HTTP traffic to pass when WAAS is enabled.
- We expect HTTP GET response times to decrease with WAAS acceleration enabled.

Results

HTTP Acceleration Branch 3 passed.

CIFS/WAFS Performance

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.

The following tests were performed:

- WAFS Configuration Verification
- CIFS Cache Hit Benchmark Branch 1
- CIFS Cache Hit Benchmark Branch 2
- CIFS Cache Hit Benchmark Branch 3
- CIFS Cache Miss Benchmark Branch 1
- CIFS Cache Miss Benchmark Branch 2
- CIFS Cache Miss Benchmark Branch 3
- CIFS Native WAN Benchmark Branch 1
- CIFS Native WAN Benchmark Branch 2
- CIFS Native WAN Benchmark Branch 3
- CIFS Verification WAE502
- CIFS Verification WAE512
- CIFS Verification WAE612

WAFS Configuration Verification

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. 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 is designed to validate the WAFS configuration for the WAAS network. The configuration of the appropriate system and services is verified. Even though the auto-discovery feature is enabled a file server is still defined for verification since in some cases this may be preferable. Then the WAFS policies are 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 Frm 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 -> wae-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 checkbox (CM -> Services -> File -> Connectivity -> wae-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: 1544Kbits/sec Minimum roundtrip delay: 80ms

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 adaptor configuration on the WAE devices by issuing the **show running-config** command.

Step 17 Verify that the WAFS Edge and WAFS Core WAEs 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 Then, 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 checkmark 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

- We expect a Core cluster to have been created.
- We expect a Core WAE deviceto 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 to list the file server as exported.
- We expect the connectivity directive to list the edge device(s).
- We expect the WAFS policies to be configured on the Core and Edge device(s).
- We expect the WAFS Accept/Transport policies to be configured and enabled.

Results

WAFS Configuration Verification passed.

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 6ms 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

- 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 failed CSCsi58809.

CIFS Cache Hit Benchmark Branch 2

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 2 was simulated to have T1 bandwidth and 17ms of latency to the file server in the Data Center.

Test Procedure

The procedure used to perform the CIFS Cache Hit Benchmark Branch 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 that WCCPv2 is running on wae-2821-branch2 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 2 by issuing the **clear statistics all** command.

Step 4 Launch the benchmark tool on a Windows client at Branch 2.

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 2 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

- 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 2 failed CSCsi58809.

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

- 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 failed CSCsi58809.

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).

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 6ms 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

- 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 failed CSCsi58809.

CIFS Cache Miss Benchmark Branch 2

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.

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).

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 2 was simulated to have T1 bandwidth and 17ms of latency to the file server in the data center.

Test Procedure

The procedure used to perform the CIFS Cache Miss Benchmark Branch 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 that WCCPv2 is running on wae-2821-branch2 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 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 2 by issuing the **clear statistics all** command.

Step 5 Launch the benchmark tool on a Windows client at Branch 2.

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 2 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

- 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 2 failed CSCsi58809.

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).

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

- 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 failed CSCsi58809.

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 6ms 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 dcb-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.

Step 12 Exit and close the Cisco WAFS Benchmark Tool.

Step 13 Reconfigure WCCPv2 on wae-3845-branch1 and dca-core-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

- 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 2

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 17ms RTT latency.

Test Procedure

The procedure used to perform the CIFS Native WAN Benchmark Branch 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 Unconfigure WCCPv2 on wae-2821-branch2 and dcb-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 Branch 2 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 2.

Step 5 Check the prepare benchmark boxs.

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.

Step 12 Exit and close the Cisco WAFS Benchmark Tool.

Step 13 Reconfigure WCCPv2 on wae-2811-branch1 and dca-core-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

- We expect opened files to take considerably longer than when WAFS/CIFS acceleration is configured.

Results

CIFS Native WAN Benchmark Branch 2 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 dcb-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.

Step 12 Exit and close the Cisco WAFS Benchmark Tool.

Step 13 Reconfigure WCCPv2 on wae-2811-branch1 and dca-core-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

- 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 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 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

- 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 failed CSCsi58809.

CIFS Verification WAE512

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 T1 connection with approximately 17ms 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 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 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 2 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

- 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 WAE512 failed CSCsi58809.

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 6ms 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 2 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

- 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 failed CSCsi58809.

CHAPTER 7

Blade Servers

HP c-Class BladeSystem

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.

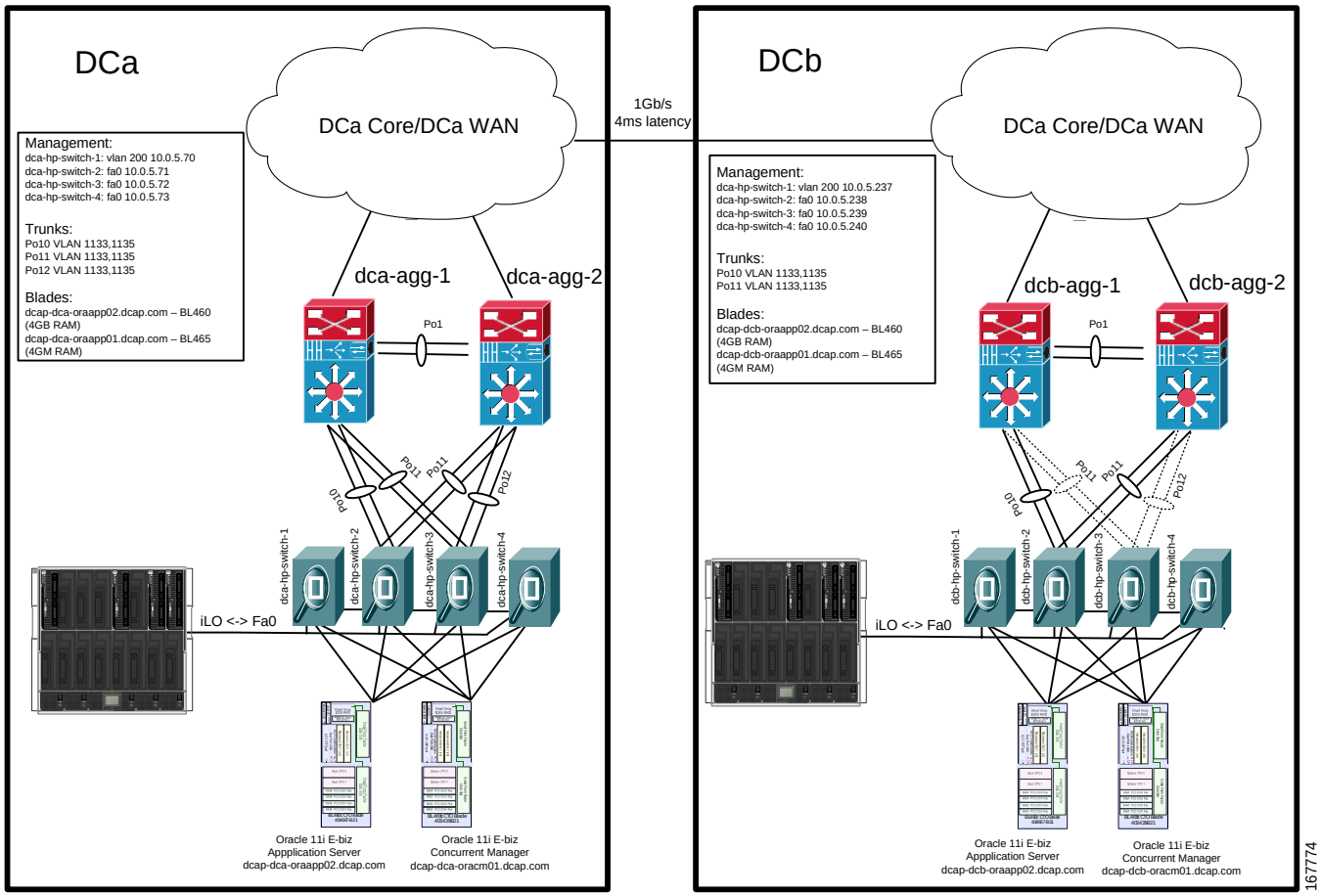
Consolidation also provides the ability to have simple to manage, easy to control, administration. With the Onboard Administrator, Integrated Lights Out (iLO), and HP Insight Control, you can manager your servers and Cisco switches by taking complete control regardless of the state of the server operating 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

Figure 7-1 shows the blade server topology configuration of the DCAP test topology.

Figure 7-1 HP Bladerservers Topology



Blade Servers Test Results Summary

Table 7-1 summarizes tests executed as part of the Cisco DCAP 3.0 testing initiative. Table 7-1 includes the feature or function tested, the section that describes the feature set the feature or function belongs to, the component tests for each feature or function, and whether the test is new in this phase of DCAP testing.

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

Note Test results are unique to technologies covered and actual scenarios in which they were tested. DCAP is designed to cover critical path areas and augment ongoing regression and systems testing.

Table 7-1 Cisco DCAP 3.0 Bladeservers Testing Summary

Test Suites	Features/Functions	Tests	Results
Baseline	Topology Baseline, page 7-6	1. Baseline Steady State	New
	Device Management, page 7-7	1. Upgrade 122(25)SEF1 to 122(35)SE	New, CSCsg83678
		2. Upgrade 122(25)SEF2 to 122(35)SE	New CSCsg83678
		3. Syslog Basic Functionality	New
		4. NTP Basic Functionality and Failover	New
		5. SNMP Trap Functionality	New
		6. SNMP MIB Walk	New
	CLI Functionality, page 7-15	1. Repeated Telnet Logins	New
		2. Repeated SSHv1 Logins	New
		3. Repeated SSHv2 Logins	New
		4. VTY Access List	New
	CLI Functionality, page 7-15	1. Parser RP via Telnet	New CSCsi72694
		2. Parser RP via SSHv1	New CSCsi72694
		3. Parser RP via SSHv2	New CSCsi72694
	Security, page 7-17	1. Malformed SNMP Polling	New
		2. Malformed SSH Packets	New
		3. NMAP Open Port Scan	New

Table 7-1 Cisco DCAP 3.0 Bladeservers Testing Summary (continued)

Test Suites	Features/Functions	Tests	Results
Baseline	Reliability, page 7-20	1. Power Cycle	New CSCsg83678
	SPAN, page 7-21 SRND: 2-4: Monitoring Protocols	1. Local SPAN 2. Remote SPAN	New New
Layer 2	Trunking, page 7-24	1. 802.1q Basic Functionality	New
	Spanning Tree, page 7-26 SRND: 2-32: Spanning Tree	1. RPVST+ Basic Functionality	News

Blade Servers DDTS Summary

Table 7-2 lists Development Defect Tracking System (DDTS) software bugs with descriptions, and comments filed by the DCAP testing team during Cisco DCAP 3.0 Bladeservers L2 3020 blade switch testing. Table 7-3 lists DDTS with descriptions encountered during Cisco DCAP 3.0 Bladeservers L2 3020 blade testing. Table 7-4 lists DDTS of interest but not encountered during Cisco DCAP 3.0 Bladeservers L2 3020 blade testing.

Table 7-2 Summary of DDTS Filed During Cisco DCAP 3.0 L2-3 Testing

DDTS	Description
CSCsi72694	Crash at show platform port-asic stats drop FastEthernet 0. Workaround: Do not issue show commands relating to FastEthernet 0 Commands: show platform port-asic stats drop FastEthernet 0, show platform port-asic mac-info FastEthernet 0, show platform port-asic mac-info FastEthernet 0, clear port-security sticky interface FastEthernet 0, clear port-security dynamic interface FastEthernet 0, clear port-security dynamic interface FastEthernet 0, clear port-security all interface FastEthernet 0

Table 7-3 Summary of DDTS Encountered During Cisco DCAP 3.0 L2-3 Testing

DDTS	Description
CSCsg83678	cbs3020: pm_start_recover: invalid operErrReason for Gi0/10

Table 7-4 Summary of DDTS of Interest During Cisco DCAP 3.0 L2-3 Testing

DDTS	Description
CSCsh60216	halberd2: SFP intf on 6K don't link up. Happens only with x6416-GE-MT module on

Blade Servers Test Cases

Functionality critical to global enterprises in Cisco DCAP 3.0 Bladeservers L2 3020 blade testing is described in the following sections. Refer to Cisco Data Center Assurance Program (DCAP) 3.0 Configurations document for test device configurations.

- Baseline, page 7-6
- Device Management, page 7-7
- CLI Functionality, page 7-15
- CLI Functionality, page 7-15
- Reliability, page 7-20
- Reliability, page 7-20

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.

The following test features were conducted:

- Topology Baseline, page 7-6

Topology Baseline

In all of DCAP testing, system resources of all of the Layer 2 blade switch 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.

The following test was performed:

- Baseline Steady State, page 7-6

Baseline Steady State

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 test follows:

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

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

Results

Baseline Steady State passed.

Device Management

Device Management tests 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 test verified that the Cisco IOS upgrade process worked correctly.

The following tests were performed:

- Upgrade 122(25)SEF1 to 122(35)SE, page 7-7
- Upgrade 122(25)SEF2 to 122(35)SE, page 7-8
- Syslog Basic Functionality, page 7-8
- NTP Basic Functionality and Failover, page 7-9
- SNMP Trap Functionality, page 7-10
- SNMP MIB Walk, page 7-11

Upgrade 122(25)SEF1 to 122(35)SE

This test verified the ability for the Cisco 3020 to be upgraded to the latest available version of code. The access layer device, dca-hp-switch-3, was upgraded from 12.2(25)SEF1 Native IOS to 12.2(35)SE to ensure that all hardware and configurations at the access layer were upgraded without issue.

Note The 12.2(25)SE1 code was issued a software advisory due to CSCsf31435: cbs3020: dualMedia SFP not linking with Cat6K SFP module.

Test Procedure

The procedure used to perform the Upgrade 122(25)SEF1 to 122(35)SE test follows:

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-3 is running the old Native Cisco IOS image using the **show version** command.

Step 3 Issue the **archive download-sw /overwrite tftp://172.18.177.132/cbs30x0-lanbasek9-tar.122-35.SE.tar** command to download, extract, and install the new image to flash.

Step 4 Issue the **reload** command on dca-hp-switch-3. Report any error messages seen during reload.

Step 5 Use the **show version** commands to verify that dca-hp-switch-3 came online successfully and that the new image is running.

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

- We expect the upgrade process 3020 platform to proceed smoothly and without error.

- We expect no CPU or memory problems.

Results

Upgrade 122(25)SEF1 to 122(35)SE passed with exception CSCsg83678.

Upgrade 122(25)SEF2 to 122(35)SE

This test verified the ability for the Cisco 3020 to be upgraded to the latest available version of code. The access layer device, dca-hp-switch-2, was upgraded from 12.2(25)SEF2 Native IOS to 12.2(35)SE to ensure that all hardware and configurations at the access layer were upgraded without issue.

Test Procedure

The procedure used to perform the Upgrade 122(25)SEF2 to 122(35)SE test follows:

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-2 is running the old Native Cisco IOS image using the **show version** command.

step2-log.xt

Step 3 Issue the **archive download-sw /overwrite tftp://172.18.177.132/cbs30x0-lanbasek9-tar.122-35.SE.tar** command to download, extract, and install the new image to flash.

Step 4 Issue the **reload** command on dca-hp-switch-2. Report any error messages seen during reload.

Step 5 Use the **show version** commands to verify that dca-hp-switch-2 came online successfully and that the new image is running.

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

- We expect the upgrade process 3020 platform to proceed smoothly and without error.
- We expect no CPU or memory problems.

Results

Upgrade 122(25)SEF2 to 122(35)SE passed with exception CSCsg83678.

Syslog Basic Functionality

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.

Test Procedure

The procedure used to perform the Syslog 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 that syslog is configured on dca-hp-switch-3 and that the designated server is 172.18.177.132 by issuing the **show running-config** command.

The DUT is configured to send logging messages to server 172.18.177.132.

Step 3 Enable the t**erminal monitor** command on the DUT and enable perform the **shutdown** and **no shutdown** commands on port-channel 11.

Debug messages will appear on the terminal.

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-3 (10.0.5.72).

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

- We expect each message that is logged to the VTY session to also be logged to the syslog server.
- We expect no CPU or memory problems.

Results

Syslog Basic Functionality passed.

NTP Basic Functionality and Failover

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: 172.18.177.132) 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 test follows:

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 172.18.177.132 (preferred) are configured as the NTP servers on the DUT and that 172.18.177.132 is synced to it as the master.

The two servers are configured on dca-hp-switch-3. The server 172.18.177.131 is (+) selected, as it is a qualified NTP server. The server 172.18.177.132 is (*) master (synced).

Step 3 On the server 172.18.177.132, 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 172.18.177.132 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 172.18.177.132 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 172.18.177.132 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

- 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 passed.

SNMP Trap Functionality

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. An SNMP trap is created by entering and leaving config mode on dca-hp-switch-3. The logging messages created on the device should be logged to the SNMP trap receiver.

Test Procedure

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-3 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

- 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 passed.

SNMP MIB 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 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 test follows:

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

- We expect no tracebacks or crashes to occur on the DUT.
- We expect no CPU or memory problems.

Results

SNMP MIB Walk 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 or Telnet, for security purposes. These tests stress the access protocols used.

The following tests were performed:

- Repeated Telnet Logins, page 7-12
- Repeated SSHv1 Logins, page 7-13
- Repeated SSHv2 Logins, page 7-13
- VTY Access List, page 7-14

Repeated Telnet Logins

This test verified that repeated Telnet logins to a Cisco 3020 switch did not impact memory or system stability. The device dca-hp-switch-3 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 test follows:

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

- We expect no system error messages resulting from the multiple, repeated Telnet login attempts.
- We expect no CPU or memory problems.

Results

Repeated Telnet Logins passed.

Repeated SSHv1 Logins

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-agg-1 did not cause any memory loss, tracebacks, or crashes. From a server, five version 1 SNMP walks were performed.

Test Procedure

The procedure used to perform the Repeated SSHv1 Logins test follows:

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-3 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

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

Results

Repeated SSHv1 Logins passed.

Repeated SSHv2 Logins

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 SSHv2 logins to a Cisco 3020 switch did not impact memory or system stability. The device dcb-hp-switch-3 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 test follows:

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-3 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

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

Results

Repeated SSHv2 Logins passed.

VTY Access List

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 (buladean and celo) 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 test follows:

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 1 using the**monitor session 1 source interface Port-channel 1 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 Gi0/22 using the **monitor session 1 destination interface Gi0/22** command.

Step 5 Clear the traffic counters on the switch 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 5 minutes.

Step 8 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 9 Look for any errors on the SPAN destination interface using the **show interfaces Gi0/22** 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

- We expect the IP access-class list to appropriately allow or deny access to the connecting host.
- We expect no CPU or memory problems.

Results

VTY Access List passed.

CLI Functionality

Parser testing 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).

The following tests were performed:

- Parser RP via Telnet, page 7-15
- Parser RP via SSHv1, page 7-16
- Parser RP via SSHv2, page 7-16

Parser RP via Telnet

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 test follows:

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

- 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 passed with exception CSCsi72694.

Parser RP via SSHv1

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 test follows:

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

- 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 passed with exception CSCsi72694.

Parser RP via SSHv2

An automated script was used to test the valid **show** and **clear** commands on dca-hp-switch-3. SSH version 2 was used as the access protocol.

Test Procedure

The procedure used to perform the Parser RP via SSHv2 test follows:

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

- 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 passed with exception CSCsi72694.

Security

Resistance 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.

The following tests were performed:

- Malformed SNMP Polling, page 7-17
- Malformed SSH Packets, page 7-18
- NMAP Open Port Scan, page 7-19

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.

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** command on dca-hp-switch-3.

The read-only password is public (default).

Step 4 Execute the two Protos traffic generation scripts on dca-hp-switch-3.

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

- We expect all DUT's not to pause indefinitely, crash, or give any tracebacks while test is being run.
- We expect no CPU or memory problems.

Results

Malformed SNMP Polling passed.

Malformed SSH Packets

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-3 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 test follows:

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-3 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-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 directed-request
tacacs-server key cisco
line vty 0 4
 transport input telnet ssh
```

Step 4 Verify the SSH server on dca-hp-switch-3 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

- We expect SSH vulnerability testing not to cause the switch to reload, pause indefinitely, or crash.
- We expect no CPU or memory problems.

Results

Malformed SSH Packets 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), 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 test follows:

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

- We expect the open ports revealed by the NMAP tool to be expected.

- We expect Telnet (TCP port 23), SSH (TCP port 22), HTTPS (443) to be open.
- On switches using the Fa0 port for management we expect port TCP port 49623.
- We expect no CPU or memory problems.

Results

NMAP Open Port Scan passed.

Reliability

Hardware reliability testing for Safe Harbor verified that hardware stays up as expected.

The following test was performed:

- Power Cycle, page 7-20

Power Cycle

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 test follows:

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-3, simulate a power failure in the HP Onboard administrator GUI. In the GUI select the Cisco Catalyst Blade Switch in Bay 3 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

- We expect the device to boot and come back online without error.
- We expect no tracebacks or crashes to occur on the DUT.
- We expect no CPU or memory problems.

Results

Power Cycle passed with exception CSCsg83678.

SPAN

The SPAN function on the Catalyst 6500 designates a port to monitor other port(s) and/or VLANs to allow packets to be forwarded to Network Analysis tools.

The following tests were performed:

- Local SPAN, page 7-21
- Remote SPAN, page 7-22

Local SPAN

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-3. 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 test follows:

Step 1 Begin background scripts to collect initial status of test network devices. Monitor memory and CPUs 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 the background test traffic for a period of 5 minutes.

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

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

Results

Local SPAN passed.

Remote SPAN

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-3. 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 test follows:

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 1140 using the **monitor session 1 destination remote VLAN 1140** command.

Step 5 Verify that interface GigabitEthernet 0/21 is configured to trunk VLAN 1140 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 the background test traffic for a period of 5 minutes.

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

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

Results

Remote SPAN passed.

Layer 2

Layer 2 feature testing for Safe Harbor involves the features:

- Trunking, page 7-24
- Spanning Tree, page 7-26

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. Table 7-5 lists and describes the five modes of trunking on Cisco switches.

Table 7-5 Trunking Modes 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.
Desirable	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.
Nonnegotiate	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.

The following test was performed:

- 802.1q Basic Functionality, page 7-24

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-3, 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 Begin sending background HTTP traffic.

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-3.

Step 6 Use the **show interfaces***interface***trunk** command to verify that the trunk between dca-agg-1 and dca-hp-switch-3 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 the background 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

- 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.

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.

The following test was performed:

- RPVST+ Basic Functionality, page 7-26

RPVST+ Basic Functionality

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 test follows:

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 10 and Blocking for Portchannel 11 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 11 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 10 by issuing the **no shutdown** command.

Step 7 Verify the STP state for Portchannel 11 has transitioned from Forwarding to Blocking and that the STP state for Portchannel 10 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

- We expect the switch to be operating in Rapid-PVST+ mode.
- We expect the switch to Forward on VLAN 1133 to one aggregation switch and Blocking on another.
- We expect the switch to 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 passed.

CHAPTER 8

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 customization, decreasing integration costs, and consolidating instances.

The centerpiece of the DCAP topology, with respect to Oracle application testing, is the configuration of Oracle 11i E-Business Suite 11.5.10.2 with Oracle 10gR2 Database 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. The 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 the application testing which includes running real application traffic using the HP/Mercury Load Runner tool. Traffic is sent to both data centers, DCa and DCb, with WAAS and without WAAS, from clients located at the three branch offices.

Note Failover and failback testing was conducted against this implementation of the Oracle application, the results of which are discussed in detail in the "Disaster Recovery" section on page 10-1.

Table 8-1 shows Cisco products leveraged in the DCAP topology for achieving key IT objectives in the areas of reliability, availability and serviceability (RAS).

Table 8-1 Cisco Products that Leverage IT RAS Objectives in DCAP 3.0

RAS Features	Cisco Products
Application Service Virtualization	GSS,CSM
Application Load Balancing	CSM,GSS
Application Optimization	WAAS
Business Resiliency	MDS,GSS
High Availability	MDS, Catalyst6500

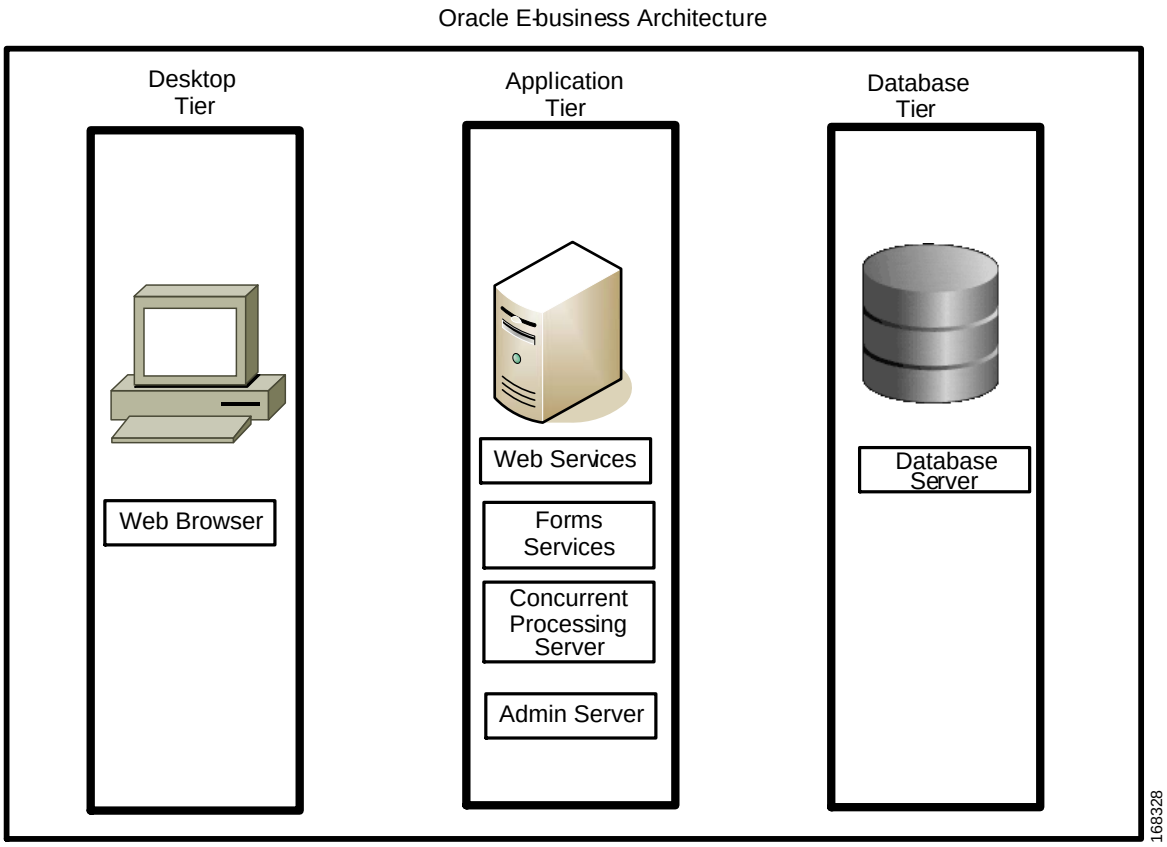
E-Business Suite Architecture

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 consists of:

1. Database Tier, which supports and manages the Oracle database.
2. Application Tier, which supports and manages various application components and is also known as Middle Tier.
3. Desktop Tier, which provides the user interface via an add-on component to a standard web browser.

Figure 8-1 shows the Oracle Application Architecture that using the logical separation of 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 8-1 Oracle E-Business Suite Architecture



Desktop Tier

The Desktop Tier represents on Internet or Intranet clients 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 Tier and Database Tier. Figure 8-1 shows the four service groups that comprise basic Application Tier for Oracle Applications.

- Web Server
- Forms Server
- Concurrent Processing Server
- Admin Server

Database Tier

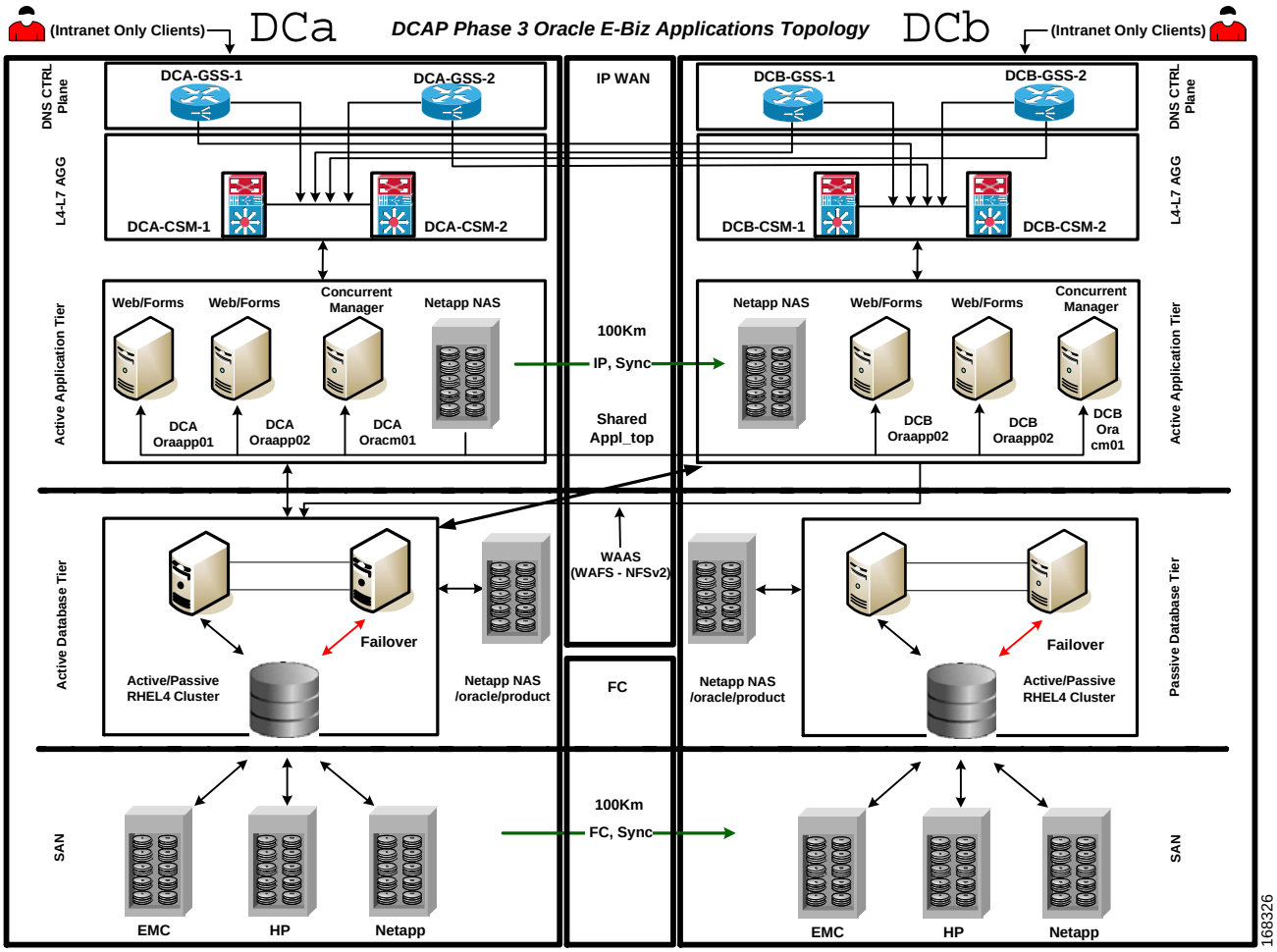
The Database Tier contains the Oracle Database Server, which stores all the data maintained by Oracle Applications. Clients in the Desktop Tier don't communicate directly with the Database (DB) Tier. Instead, servers in the Application Tier communicate with the DB servers to process client requests.

DCAP Oracle E-Business Topology

Figure 8-2 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 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 the 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 the logical tiers of E-Business Suite.

Figure 8-2 provides an overview of the DCAP 3.0 Oracle E-Business Suite topology, including the major components.

Figure 8-2 Cisco DCAP 3.0 Oracle E-Business Topology Overview



Desktop Tier

Intranet clients shown in the topology represent the Desktop Tier. Clients for the Oracle Application are located in three branch offices. Table 8-2 shows 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 8-2 Cisco DCAP 3.0 Branch Configurations

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 and Oracle Forms port 9000 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 transactions.

Aggregation Tier

All four GSS's (two GSS's 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's were authoritative for the domain wwwin-oefin.gslb.dcap.com. All four GSS's provided health checks for the Oracle application which was virtualized on each CSM at each of the two data centers. The GSS's maintained the health of the Oracle applications running behind each of the CSM's. A client DNS request arrives at each of the four GSS's 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's, the 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 either DCa or DCb. At the time that the DNS query arrives at one of the four GSS's the GSS's already are aware which VIP's at each data center are alive and available. 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 application's health which the 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 one of the CSM's at either DCa or DCb. The different types of load balancing for the GSS DNS rule on the GSS's are as follows: Round Robin, Weighted Round Robin, Ordered List, Least Loaded and Hashed. Our tests used both Round Robin and Weighted Round Robin.

For additional details on how GSS is configured, refer to the "Global Site Selector (GSS)" section on page 5-1.

The CSM's that are connected into each of the aggregation switches at both DCa and DCb provide a level of virtualization and load balancing for the Oracle E-Business Suite of Applications for which it is providing services for. Each CSM has three specific Vservers configured in order to provide services for the Oracle E-Business Suite of applications. The first Vserver is a type redirect.

```
vserver WWWIN-REDIRECT
  virtual 101.40.1.51 tcp www
  serverfarm 80-TO-8000
  persistent rebalance
  no inservice
```

The Vserver above takes a client request that is destined for TCP port 80 (HTTP Traffic) and issues an HTTP 302 redirect for TCP port 8000. This is the first Vserver matched for incoming HTTP requests.

```
vserver WWWIN-OEFIN
  virtual 101.40.1.51 tcp 8000
  vlan 301
  serverfarm ORACLE-ALL
  advertise active
  sticky 30 group 30
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  domain dca-csm-1
  inservice
```

TheVserver above receives HTTP requests (wwwin-oefin.gslb.dcap.com) on TCP port 8000 for which the first Vserver redirected the client to. This Vserver matches a client request that arrives at the CSM on TCP port 8000 and load balances the request to one of the servers in the serverfarm (ORACLE-ALL). This Vserver also creates a sticky entry into the sticky database based on the client's source IP address.

```
vserver WWWIN-OEFIN-9K
  virtual 101.40.1.51 tcp 9000
  vlan 301
  serverfarm ORACLE-ALL
  advertise active
  sticky 30 group 30
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  domain dca-csm-1
inservice
```

The above Vserver receives HTTP requests on TCP port 9000 for which the Oracle Forms applications redirected the client to. This Vserver will match a client request that arrives at the CSM on TCP port 9000 and ensure that the client will be load balanced to the same real servers as the Vserver WWWIN-OEFIN. .

All four GSS's communicate with the CSM's at both Data Centers. All four GSS's must be able to reach the CSM's at both DCa and DCb in order to understand their health and availability. This function is called a keepalive. The following KAL-AP method is used in the DCAP topology.

- KAL-AP – Uses a UDP transport where the GSS interrogates a CSS/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.

KAL-AP by TAG was used for the keepalive type between all four GSS's and the CSM's at each Data Center.

It is extremely important to understand the different TCP and UDP ports that the GSS's use for keepalive functionality in order to help plan network topologies and positioning of the GSS's. In the topology for both DCa and DCa, NAT was not a factor, however the proper ports were allowed through the FW's for all keepalive/probe types.

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 and the probes are retried.
- Failed—The real server fails to reply after a specified number of consecutive retries. You are notified and the 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 HTTP payload includes an HTTP Header named "I_AM_CSM" in order to aid in troubleshooting and validation.

The two health probes used on the CSM during DCAP testing are as follows:

An HTTP probe 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 sending 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 5
  failed 2
  port 8000

dca-agg-1#show mod csm 2 probe name oracle detail
probe           type    port  interval retries failed  open   receive
---------------------------------------------------------------------
ORACLE          http    8000  5        3       2       10     10
 Probe Credentials:
  Username: sysadmin             Passwd: sysadmin
 Probe Request:  GET        /oa_servlets/AppsLogin
 HTTP Headers:
  I_AM_CSM
 Expected Status Codes:
  302
```

The second probe, shown below, creates a TCP probe assigned to the database server named “ORACLE-FORM”.
This probe creates a TCP 3-way handshake to port 9000. Upon receiving acknowledgement that TCP port 9000 on the database server is responding with a TCP SYN/ACK, the CSM assumes the application is operational and makes the server available as a resource.

```
probe ORACLE-FORMS tcp
  interval 5
  retries 2
  port 9000

dca-agg-1#show mod csm 2 probe name oracle-forms detail
probe           type    port  interval retries failed  open   receive
---------------------------------------------------------------------
ORACLE-FORMS    tcp     9000  5        2       300     10
```

Application Tier

Three application hosts DCa-Oraapp01 (Penguin), DCa-Oraapp02 and DCa-Oracm01 (HP Blade Servers) make up the application Tier in DCa. DCa-Oraapp01 and DCa-Oraapp02 are configured to provide front-end connectivity functions servicing Web and Forms services. Application hosts are configured behind CSM to provide load balancing capabilities while at the same time providing high availability for services within the data center. Host DCa-oracm01 provides services to run Concurrent Manager/Batch jobs. The setup is similar in DCb where hosts DCb-oraapp01 and DCb-Oraapp02 provide Web and forms services and DCb-oracm01 is configured as failover Concurrent Manager Server since the Parallel Concurrent Processing feature is currently not enabled.

The CSM, which resides in the Aggregation Layer at each data center, provides at each data center provides load balancing between the two 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 Layer 2 adjacency between the data centers and no advanced capabilities like route health injection are being used. In this 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 CSM VIP. Table 8-3 highlights changes required in the context file for each of the Application hosts to accommodate integration of the CSM with E-Business Suite.

Table 8-3 Application Context File Changes to Accommodate Hardware Load Balancer CSM

Variable Name	Current Values	Changed Value
s_webentryhost	dcap-dca-oraapp01 - DCa App Node1 dcap-dca-oraapp02 - DCa App Node2 dcap-dcb-oraapp01 - DCb App Node1 dcap-dcb-oraapp02 - DCb App Node 2	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-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://wwwin-oefin.gslb.dcap.com/oa_servlets/AppsLogin

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 the Application Tier tech stack file system. All the application tier files are installed on a single NetApp filer cluster volume (/apps/oefin) located in data center A and mounted using NFS V2 over TCP across all the Application hosts in DCa and DCb. To enable failover to data center B, the 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 a key component for an 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 users.

- 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 minimizes the duration of planned maintenance downtime.

Note For more information on how to enable shared appl_top please refer to, Note ID: 233428.1 http://metalink.oracle.com

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: “Socket Mode” is enabled.When using the CSM to load balance to both the application servers for Oracle Forms it is required that sticky is enabled on the CSM. Sticky must be enabled on the CSM to ensure that the user that was load balanced to the application server on port 8000 is load balanced to the same server for port 9000 (the TCP port used for Oracle forms). In the testing, the method of sticky used was “sticky source IP address”.

Key drivers for enabling the Socket Mode in the topology are:

- To reduce the consumption of resources by the JVM’s on the Application Tier
- To reduce network traffic since the servlet mode uses the HTTP protocol and each transaction requires the exchange of cookies and HTTP headers
- The customer’s network topology requires Desktop Tier clients to access the Forms server directly.

Database Tier

The Database Tier consists of two RedHat Enterprise Linux 4 update 4 active/passive database clusters, one in each data center. The two hosts in each cluster access the Oracle executable code from a Network Appliance NAS filer in each data center (each cluster has a separate filer). The database instance, called OEFIN, is normally located on SAN storage in data center A from either EMC, HP, and Network Appliance. 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 fiber channel link to ensure an identical copy of the database is available for failover in data center B.

DCAP Oracle E-Business Environment

The following list summarizes hardware and software used in the DCAP E-Business environment.

Hardware

Application Tier

- 2 Penguin Servers ALTUS 1600
- 2 HP Blade Servers BL460c with Dual Intel® Xeon™ 1.6Ghz CPU and 4GB of RAM
- 2 HP BL465c with Dual AMD Opteron 1.8GHz CPU and 4GB of RAM

Database Tier

- 4 HP Compaq DL380 G4s with an Intel® Xeon™ 3.20 GHz CPU and 4 GB of RAM.

Software

Oracle E-Business Suite

- Oracle11i – 11.5.10.2

Techstack Patch Level

- ATG RUP4

Oracle Database

- Oracle 10gR2

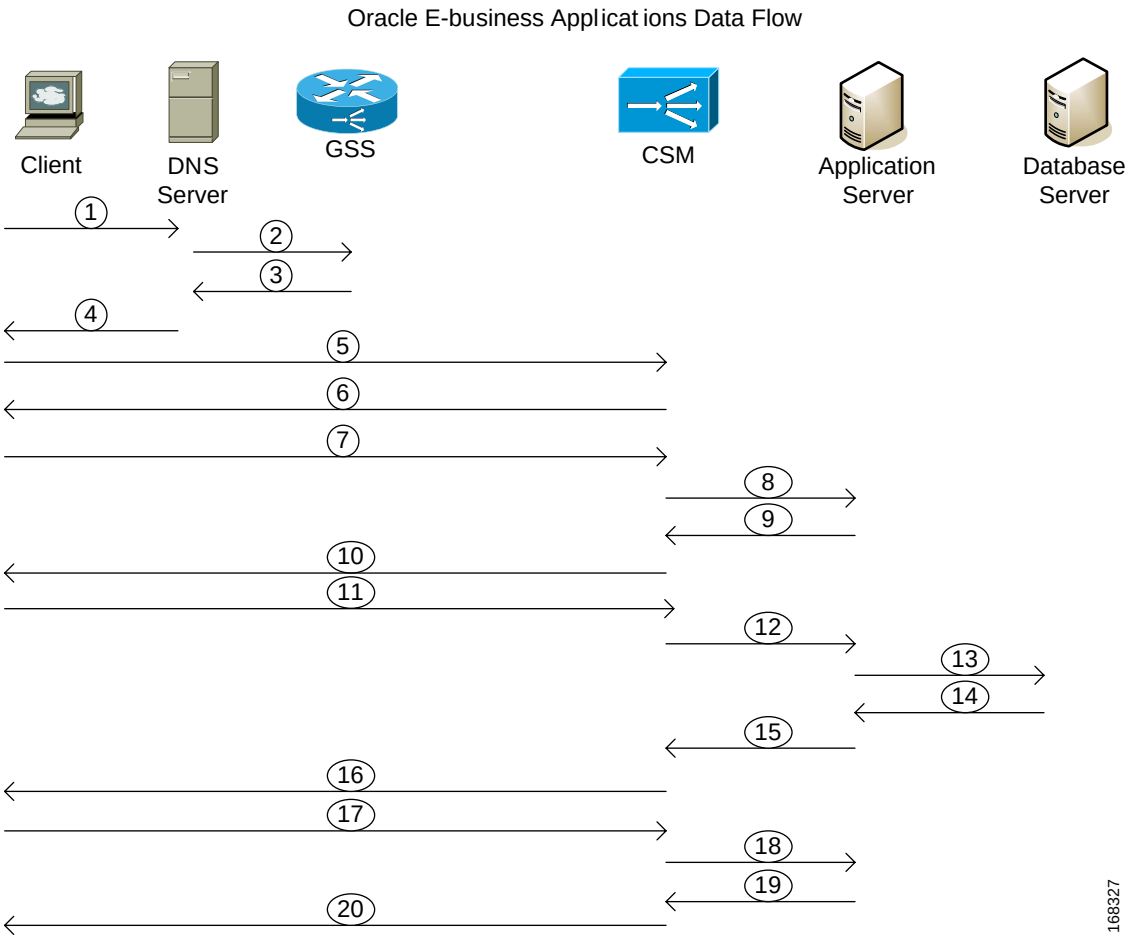
Operating System

- RHEL4 Update 4 32-bit for Application Tier
- RHEL4 Update 4 64-bit for Database Tier

Application Traffic Flow

Figure 8-3 provides an overview of the Application data flow from a Branch Client to the E-Business Suite application showing 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 8-3 Cisco DCAP 3.0 Oracle Application Data Flow



Step 1 Client DNS requests a query to wwwin-oefin.gslb.dcap.com to Branch Name Server

Step 2 Branch Name Server NS Forwards wwwin-oefin.gslb.dcap.com to GSS.

Step 3 GSS Authoritative Answers VIP's 101.40.1.51 (DCa) or 201.40.30.51 (DCb) to Branch Name Server.

Step 4 Branch Name Server Cached response VIP from DCa or DCb for a TTL of 5sec to Client.

Step 5 Client sends http://wwwin-oefin.gslb.dcap.com request to port 80 on CSM.

Step 6 CSM sends HTTP 302 Redirect to http://wwwin-oefin.gslb.dcap.com:8000 to Client.

Step 7 Client sends http://wwwin-oefin.gslb.dcap.com request to port 8000 on CSM.

Step 8 CSM load balances client request (GET) to one of the Application servers behind CSM and creates the sticky entry.

Step 9 Application server responds back to CSM with the server response (HTML).

Step 10 CSM sends the response back (HTML) to the Client. This is the Oracle E-Business Login Page.

Step 11 Client clicks on E-Business Login page link and submits the credentials for userid /password.

Step 12 CSM forwards request to Application server.

Step 13 Credentials are sent from Application server to Database Host via TCP connection. Destination IP address is of the DB server and port is 1521 where DB listener is configured.

Step 14 Credentials are validated against Fnd_user table in DB and resulting page with appropriate user responsibilities is sent back to the Application server.

Step 15 Application server forwards resulting HTTP response back to CSM.

Step 16 CSM forwards the HTTP server response back to Client.

Step 17 Client initiates http request to wwwin-oefin over the port (9000) to access Forms portion of the Application.

Step 18 CSM forwards http request to the same Application Server that was chosen from. Step 8 is based on the sticky table in CSM on port 9000. The request is handled by the forms listener configured on port 9000.

Step 19 Application server sends the response for port 9000 back to CSM.

Step 20 CSM sends the request back to client. At this point Oracle forms are opened in user's browser through the Java client applet running in the browser. When the user initiates an action in the applet such as entering data or clicking a field, data is passed to the forms server running on Application host. If necessary database tier is contacted (data is sent through TCP port 1521) for any data that is not cached on the Application host or for data processing.

Note The steps listed from 13 through 16 are repeated for data retrieval.

Testing Summary

Cisco DCAP 3.0 Oracle Applications tests fall into two major categories: Baseline Functionality to verify configuration and functionality of Oracle E-Business Suite integration with GSS,CSM, Active/Active hybrid mode and WAAS optimizations. Application traffic is generated by clients in the three branch offices to both data centers with WAAS and without WAAS. Details about Oracle application failover and failback testing conducted as part of the data center failover testing can be found in the "Disaster Recovery" section on page 10-1.

HP's Mercury Load Runner tool is leveraged to simulate application traffic. The Load runner environment has one controller located in DCa and three generators located at branch offices. Five business processes, Create_Project_forms, CreateInovice, CRM_Manage_Role, DCAP_Receivables, and iProcurement (comprised of Oracle Forms and HTTP functionality) are identified to simulate real time traffic. Details on the 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.

1. 10 Simultaneous Users test (for each of 3 branch generators for DCa and DCb separately with WAAS turned on and off)
2. 150 Simultaneous Users (global test with both DCa and DCb in the configuration with WAAS turned on and off)
3. 150 Simultaneous Users (for failover testing)

Summary Results

This section summarizes results from testing conducted at individual branches to each of the data centers. Graphs provide comparison on average transaction response times on non-optimized WAN without WAAS and improvements provided by the WAAS solution. Please see the detailed test results for more information. Detailed optimization statistics from WAAS can also be found in test results.

Figure 8-4 provides a comparison of Average Transaction Response times from Branch1 to DCa with WAAS and without WAAS

Figure 8-5 provides a comparison of Average Transaction Response times from Branch2 to DCa with WAAS and without WAAS

Figure 8-6 provides a comparison of Average Transaction Response times from Branch3 to DCa with WAAS and without WAAS

Figure 8-7 provides a comparison of Average Transaction Response times from Branch1 to DCb with WAAS and without WAAS

Figure 8-8 provides a comparison of Average Transaction Response times from Branch2 to DCb with WAAS and without WAAS

Figure 8-9 provides a comparison of Average Transaction Response times from Branch3 to DCb with WAAS and without WAAS

Figure 8-4 Cisco DCAP 3.0 Branch1 to DCa WAAS Comparison

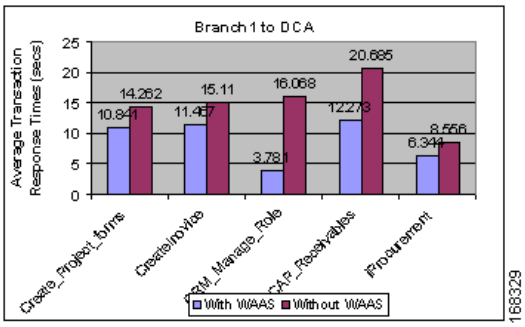


Figure 8-5 Cisco DCAP 3.0 Branch2 to DCa WAAS Comparison

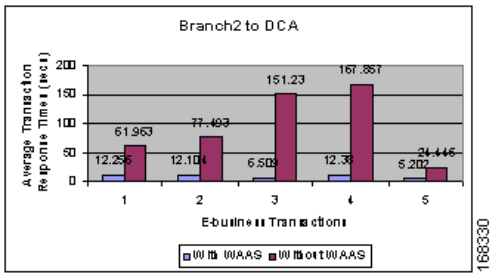


Figure 8-6 Cisco DCAP 3.0 Branch3 to DCa WAAS Comparison

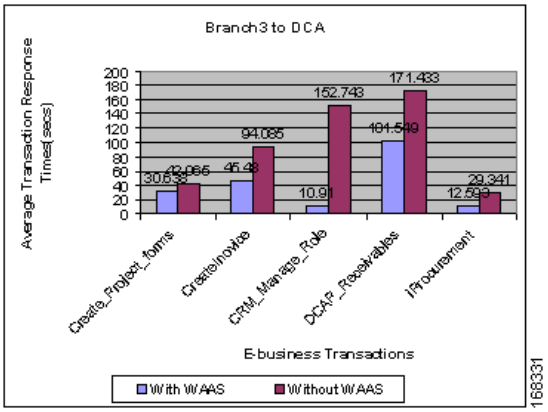


Figure 8-7 Cisco DCAP 3.0 Branch1 to DCb WAAS Comparison

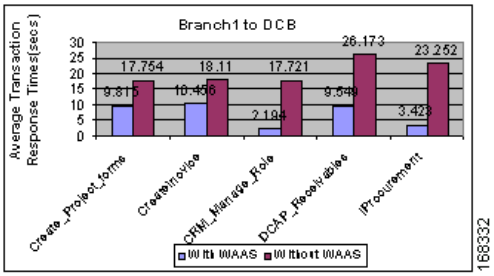


Figure 8-8 Cisco DCAP 3.0 Branch2 to DCb WAAS Comparison

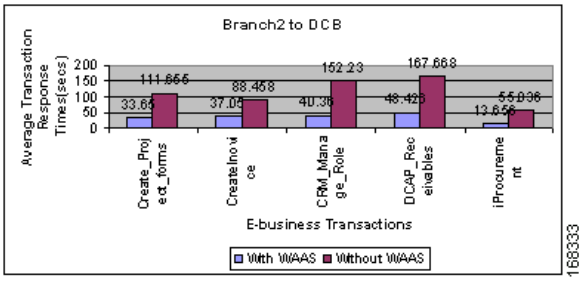
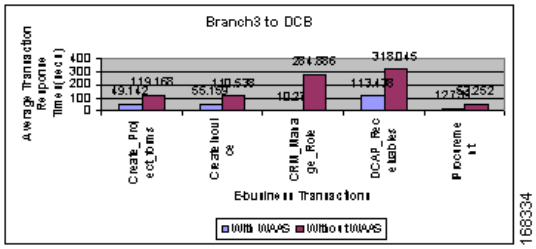


Figure 8-9 Cisco DCAP 3.0 Branch3 to DCb WAAS Comparison



Oracle Failover/Failback Summary

Table 8-4 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 the test are covered in the "Disaster Recovery" section on page 10-1.

Table 8-4 Oracle Failover/Failback Test Results

	Failover		Failback	
Vendor	**RPO**	**RTO**	**RPO**	**RTO**
EMC	0	33 min	0	97 min
HP	0	148 min *	0	20 min
NetApp	0	24 min	0	20 min

Note Actual failover time is 43mins excluding delays due to unavailability of DBa. Refer to the "Disaster Recovery" section on page 10-1.

Oracle Test Results Summary

Table 8-5 summarizes tests executed as part of the Cisco DCAP 3.0 testing initiative. Table 8-5 includes the feature or function tested, the section that describes the feature set the feature or function belongs to, the component tests for each feature or function.

Note Test results are unique to technologies covered and actual scenarios in which they were tested. DCAP is designed to cover critical path areas and augment ongoing regression and systems testing.

Table 8-5 Cisco DCAP 3.0 Oracle Testing Summary

Test Suites	Features/Functions	Tests	Results
Oracle E-Business Suite	E-Biz Configuration Validation, page 8-17	1. Oracle E-Business Applications—Environment Validation	
	E-Biz Branches to DCa, page 8-21	1. Oracle Apps Traffic from Branch 1 to DCa without WAAS 2. Oracle Apps Traffic from Branch 2 to DCa without WAAS 3. Oracle Apps Traffic from Branch 3 to DCa without WAAS	
	E-Biz Branches to DCa with WAAS, page 8-28	1. Oracle Apps Traffic from Branch 1 to DCa with WAAS 2. Oracle Apps Traffic from Branch 2 to DCa with WAAS 3. Oracle Apps Traffic from Branch 3 to DCa with WAAS	
	E-Biz Branches to DCb, page 8-34	1. Oracle Apps Traffic from Branch 1 to DCb without WAAS 2. Oracle Apps Traffic from Branch 2 to DCb without WAAS 3. Oracle Apps Traffic from Branch 3 to DCb without WAAS	
	E-Biz Branches to DCb with WAAS, page 8-41	1. Oracle Apps Traffic from Branch 1 to DCb with WAAS 2. Oracle Apps Traffic from Branch 2 to DCb with WAAS 3. Oracle Apps Traffic from Branch 3 to DCb with WAAS	
Global E-Business Suite Across Data Centers		1. Global Distribution of Oracle Apps Traffic without WAAS 2. Global Distribution of Oracle Apps Traffic with WAAS	

Oracle DDTS Summary

Table 8-6 lists Development Defect Tracking System (DDTS) software bugs with descriptions, and comments filed by the DCAP testing team during Cisco DCAP 3.0 Oracle testing. Table 8-7 lists DDTS with descriptions encountered during Cisco DCAP 3.0 Oracle testing.

Table 8-6 Summary of DDTS Filed During Cisco DCAP 3.0 Oracle Testing

DDTS	Description
N/A	

Table 8-7 Summary of DDTS Encountered During Cisco DCAP 3.0 Oracle Testing

DDTS	Description
N/A	

Oracle Test Cases

Functionality critical to global enterprises in Cisco DCAP 3.0 Oracle testing is described in the following sections. Refer to Cisco Data Center Assurance Program (DCAP) 3.0 Configurations document for test device configurations.

Oracle 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 customization, decreasing integration costs, and consolidating instances.

The following test features were conducted:

E-Biz Configuration Validation

E-Biz configuration validation was created to verify the basic configuration and functionality of the Oracle E-Business Suite. It validates functionality of Oracle Forms and HTML modules.

The following test was performed:

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 verified configuration validation of applications in four categories:

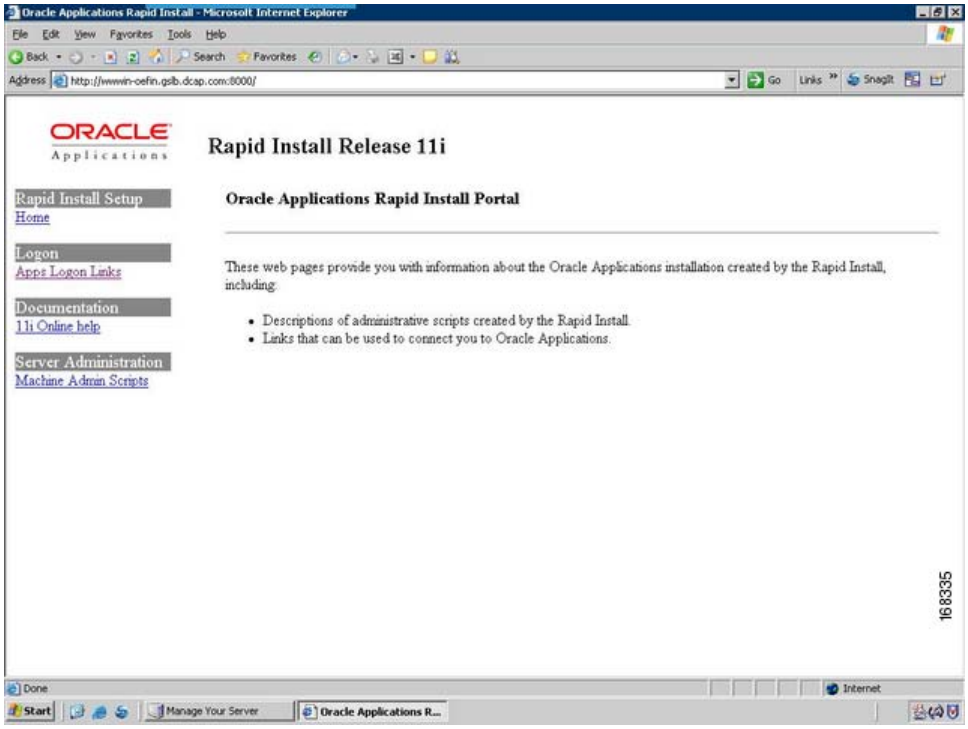
- iAS
- Oracle Applications Framework
- RDBMS
- Environment

Test Procedure

The procedure used to perform the Oracle E-Business Applications—Environment Validation test follows:

Step 1 Verify all the configuration related 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 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 2 Validate web server running on Application hosts and able to render static html.



Step 3 Validate the profile options for the following 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 4 Verify the guest user information by running the following sql to validate the RDBMS setup.

```
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 and the ICX_PARAMETERS table by running the following SQL:

```
select count(*) from icx_parameters;
```

Step 5 Verify access to Oracle Forms through the application. Follow the steps and verify that you can successfully access forms.

a. Login to homepage http://wwwin-oefin.gslb.dcap.com:8000/ and click on Apps Logon Link.

b. Click on ebusiness home page.

c. Login using user id: sysadmin and password: sysadmin.

d. Click System Administrator on Responsibilities navigation pane on the left.

e. Click Requests under Concurrent.

Step 6 Validate the concurrent manager setup, submit a batch request and validate the log and report file by viewing the results. Perform the following steps.

a. Login to homepage http://wwwin-oefin.gslb.dcap.com and click on Apps Logon Link.

b. Click on ebusiness home page.

c. Login using user id: sysadmin and password: sysadmin.

d. Click System Administrator on Responsibilities navigation pane on the left.

e. Click Requests under Concurrent.

f. Click on Submit new request.

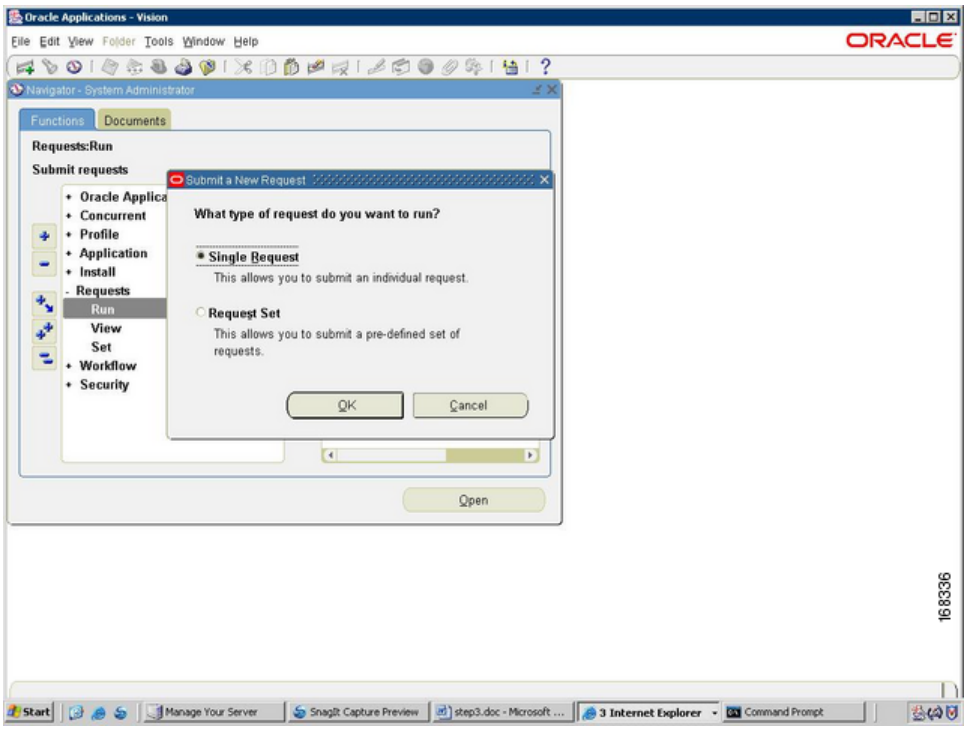
g. Type Active Resp% and click find.

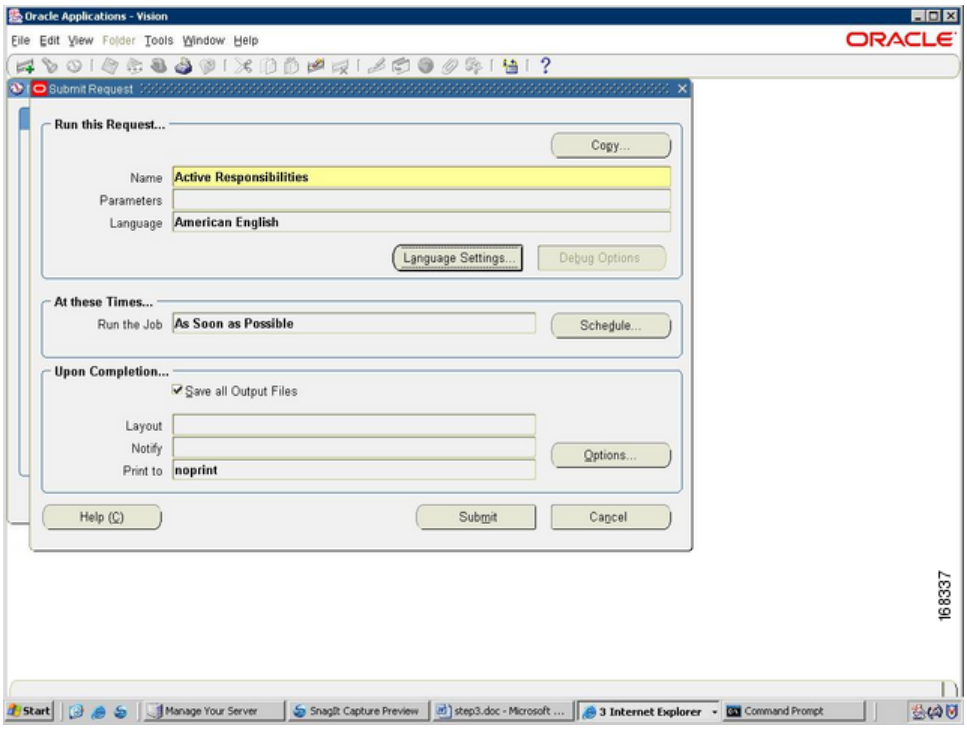
h. Select Active Responsibilities and hit submit.

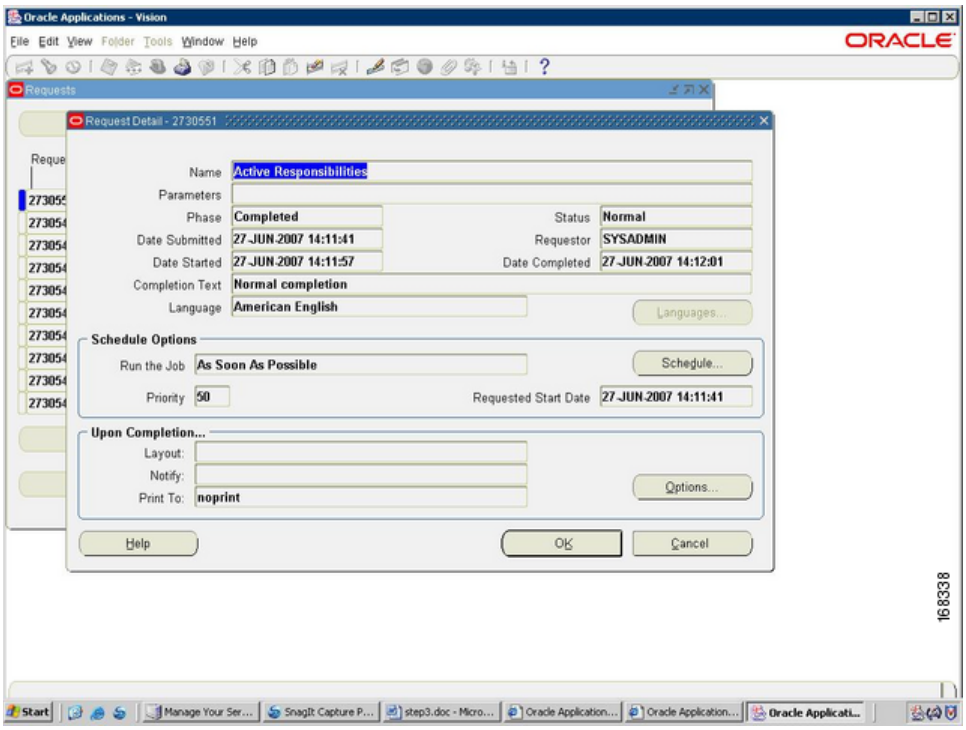
i. Click View Running requests.

j. Identify the request you just submitted and wait for it to complete.

k. View the details.


Oracle Applications - Vision
File Edit View Folder Tools Window Help
ORACLE
Navigator - System Administrator
Functions
Documents
Requests:Run
Submit requests
Oracle Applica
Concurrent
Profile
Application
Install
Requests
Run
View
Set
Workflow
Security
Submit a New Request
What type of request do you want to run?
Single Request
This allows you to submit an individual request.
Request Set
This allows you to submit a pre-defined set of requests.
OK
Cancel
Open
168336
Start
Manage Your Server
SnagIt Capture Preview
step3.doc - Microsoft ...
3 Internet Explorer
Command Prompt


Oracle Applications - Vision
File Edit View Folder Tools Window Help
ORACLE
Submit Request
Run this Request...
Copy...
Name
Active Responsibilities
Parameters
Language
American English
Language Settings...
Debug Options
At these Times...
Run the Job
As Soon as Possible
Schedule...
Upon Completion...
Save all Output Files
Layout
Notify
Print to
noprint
Options...
Help (C)
Submit
Cancel
168337
Start
Manage Your Server
SnagIt Capture Preview
step3.doc - Microsoft ...
3 Internet Explorer
Command Prompt



Expected Results

- We expect TNS connectivity is valid from all the Application hosts.
- 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.

E-Biz Branches to DCa

E-Biz Branches to DCa verifies configuration and functionality of Oracle E-Business Application deployment in Data Center A. These tests validate integration of the Application with GSS and CSM. Application traffic is simulated using Mercury Load runner from all the 3 branch clients. WAAS is disabled and GSS was configured to be authoritative for the wwwin-oefin domain which resolved client requests to the VIP physically located in DCa.

The following tests were performed:

- Oracle Apps Traffic from Branch 1 to DCa without WAAS, page 8-22
- Oracle Apps Traffic from Branch 2 to DCa without WAAS, page 8-24
- Oracle Apps Traffic from Branch 3 to DCa without WAAS, page 8-26

Oracle Apps 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 CSM load balanced the connections across the application hosts as expected.

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 Apps Traffic from Branch 1 to DCa without WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 3 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 4 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps determine the percentage of transactions that met predefined response times.

Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev.
	Average Transac	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.026	0.071	0.007	0.032
	Average Transac	1	Create_Project_forms_Click_Copy_To	0.135	0.207	0.385	0.16	0.103
	Average Transac	1	Create_Project_forms_Click_find	0.493	0.561	0.657	0.547	0.06
	Average Transac	1	Create_Project_forms_click_open	0.119	0.174	0.265	0.137	0.065
	Average Transac	1	Create_Project_forms_Click_project_link	5.707	6.853	12.259	11.181	2.698
	Average Transac	1	Create_Project_forms_Click_project_name	0.356	0.449	0.582	0.436	0.083
	Average Transac	1	Create_Project_forms_Click_project_vision_service	0.641	0.677	0.995	0.978	0.142
	Average Transac	1	Create_Project_forms_EBusiness_Links	0.121	0.193	0.231	0.226	0.051
	Average Transac	1	Create_Project_forms_Login	0.196	0.754	1.527	0.539	0.564
	Average Transac	1	Create_Project_forms_Logout	0.061	0.106	0.186	0.073	0.056
	Average Transac	1	Create_Project_forms_select_Change_Status	0.232	0.324	0.451	0.269	0.093
	Average Transac	1	Create_Project_forms_select_Customer_name	0.437	0.601	0.739	0.626	0.125
	Average Transac	1	Create_Project_forms_select_organization	0.328	0.548	0.679	0.439	0.239
	Average Transac	1	Create_Project_forms_select_PM	0.328	0.465	0.629	0.437	0.124
	Average Transac	1	CreateInvoice_Click_Apps_Logon_Links	0	0.069	0.02	0.017	0.069
	Average Transac	1	CreateInvoice_Click_Distributions	0.479	0.706	0.947	0.826	0.188
	Average Transac	1	CreateInvoice_Click_EBusiness_Links	0.139	0.334	0.666	0.284	0.199
	Average Transac	1	CreateInvoice_click_Entry_Invoices	4.72	9.416	14.318	6.269	4.219
	Average Transac	1	CreateInvoice_click_Payables_Vision_Operations	0.26	0.574	0.928	0.535	0.274
	Average Transac	1	CreateInvoice_Login	0.201	0.36	0.584	0.371	0.142
	Average Transac	1	CreateInvoice_logout	0.056	0.167	0.286	0.241	0.099
	Average Transac	1	CreateInvoice_select_account	0.477	0.864	1.297	1.094	0.341
	Average Transac	1	CreateInvoice_select_Save	1.417	1.593	1.779	1.755	0.175
	Average Transac	1	CreateInvoice_select_Supplier_Name	0.068	0.113	0.198	0.073	0.06
	Average Transac	1	CreateInvoice_select_taxcode	0.259	0.389	0.762	0.269	0.215
	Average Transac	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.029	0.059	0.057	0.029
	Average Transac	1	CRM_Manage_Role_Click_CRM_HTML_Admnistr	0.173	0.480	1.12	0.456	0.381
	Average Transac	1	CRM_Manage_Role_Click_Setup_Home	0.159	0.786	1.513	1.298	0.624
	Average Transac	1	CRM_Manage_Role_Click_Update	0.099	0.166	0.123	0.103	0.01
	Average Transac	1	CRM_Manage_Role_Click_User_Maintenance	0.056	0.215	0.526	0.065	0.219
	Average Transac	1	CRM_Manage_Role_Click_Users	0.055	0.365	0.656	0.555	0.25
	Average Transac	1	CRM_Manage_Role_EBusiness_Links	0.027	0.261	0.768	0.219	0.393
	Average Transac	1	CRM_Manage_Role_Login	0.154	0.369	0.61	0.162	0.213
	Average Transac	1	CRM_Manage_Role_Logout	0.094	0.112	0.159	0.098	0.028
	Average Transac	1	CRM_Manage_Role_Search	0.079	0.105	0.143	0.093	0.027
	Average Transac	1	CRM_Manage_Role_Select_Role	0.093	0.134	0.177	0.141	0.038
	Average Transac	1	CRM_Manage_Role_Select_Test_user	0.067	0.077	0.096	0.068	0.014
	Average Transac	1	DCAP_Receivables_Click_LineItems_EnterDet	1.491	1.712	1.88	1.741	0.14
	Average Transac	1	DCAP_Receivables_Click_Transactions	5.302	19.657	28.677	24.992	10.261
	Average Transac	1	DCAP_Receivables_EBusiness_Links	0.024	0.091	0.13	0.117	0.047
	Average Transac	1	DCAP_Receivables_EnterTransDetails_Save	0.432	0.518	0.651	0.54	0.087
	Average Transac	1	DCAP_Receivables_LineItems_Save	0.336	0.367	0.442	0.349	0.044
	Average Transac	1	DCAP_Receivables_Login	0.173	0.248	0.356	0.215	0.079
	Average Transac	1	DCAP_Receivables_LogonLinks	0	0.033	0.099	0	0.047
	Average Transac	1	DCAP_Receivables_Logout	0.062	0.132	0.272	0.063	0.099
	Average Transac	1	DCAP_Receivables_ShipTo	0.773	0.946	1.249	0.883	0.18
	Average Transac	1	DCAP_Receivables_Vision_Operation	0.302	0.650	0.975	0.797	0.254
	Average Transac	1	Procurement_AppsLogonLinks	0	0.013	0.026	0.026	0.013
	Average Transac	1	Procurement_Choose	0.42	1.276	2.133	2.133	0.856
	Average Transac	1	Procurement_Click_AddToCart	4.291	5.043	5.925	4.913	0.673
	Average Transac	1	Procurement_Click_AnkleSupport	0.258	0.863	1.468	1.468	0.605
	Average Transac	1	Procurement_Click_Categories	0.139	0.31	0.592	0.2	0.201
	Average Transac	1	Procurement_Click_Ergonomic_Supplies	0.127	0.314	0.661	0.333	0.217
	Average Transac	1	Procurement_Click_ViewCart	0.063	0.187	0.331	0.259	0.111
	Average Transac	1	Procurement_Delete_Item	0.061	0.093	0.143	0.074	0.036
	Average Transac	1	Procurement_EBusinessHomepage	0.052	0.169	0.164	0.164	0.056
	Average Transac	1	Procurement_Homepage	0	0.044	0.088	0.088	0.044
	Average Transac	1	Procurement_Logon	0.156	0.454	0.753	0.753	0.298
	Average Transac	1	Procurement_Logout	0.069	0.093	0.137	0.072	0.032
	RunningVusers	1	Run	0	3.727	10	3	3.387

168339

Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	Create_Project_forms_Click_Apps_Logon_Links	0.007	0.023	0.071	0.007	0.020
	1	Create_Project_forms_EBusiness_Links	0.121	0.203	0.295	0.229	0.065
	1	Create_Project_forms_Login	0.196	0.7	1.527	0.666	0.506
	1	Create_Project_forms_Click_project_vision_service	0.641	0.677	0.995	0.978	0.142
	1	#Procurement_Homepage	0	0.044	0.146	0.031	0.06
	1	Create_Project_forms_Click_project_link	5.707	6.853	12.259	11.181	2.698
	1	Create_Project_forms_Click_project_name	0.356	0.449	0.582	0.436	0.083
	1	Create_Project_forms_Click_find	0.493	0.561	0.657	0.547	0.06
	1	Create_Project_forms_Click_Copy_To	0.135	0.207	0.385	0.16	0.103
	1	Create_Project_forms_select_PM	0.328	0.506	0.799	0.458	0.177
	1	Create_Project_forms_select_Customer_name	0.437	0.607	0.769	0.739	0.149
	1	Create_Project_forms_select_organization	0.328	0.631	0.993	0.764	0.264
	1	Create_Project_forms_click_open	0.119	0.197	0.388	0.143	0.111
	1	Create_Project_forms_select_Change_Status	0.232	0.356	0.671	0.269	0.163
	1	Create_Project_forms_Logout	0.058	0.095	0.186	0.073	0.053
	1	#Procurement_AppsLogonLinks	0.008	0.017	0.044	0.008	0.016
	1	#Procurement_EBusinessHomepage	0.045	0.169	0.169	0.14	0.059
	1	#Procurement_Logon	0.146	0.454	1.292	0.213	0.484
	1	CreateInvoice_Click_Apps_Logon_Links	0	0.01	0.02	0.02	0.01
	1	CreateInvoice_Click_EBusiness_Links	0.139	0.334	0.666	0.284	0.199
	1	CreateInvoice_Login	0.201	0.36	0.584	0.371	0.142
	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.03	0.059	0.059	0.029
	1	CRM_Manage_Role_EBusiness_Links	0.027	0.261	0.768	0.219	0.303
	1	CRM_Manage_Role_Login	0.154	0.384	0.95	0.37	0.283
	1	#Procurement_Choose	0.363	1.276	2.794	1.471	0.977
	1	DCAP_Receivables_LogonLinks	0	0.05	0.114	0.084	0.05
	1	DCAP_Receivables_EBusiness_Links	0.024	0.1	0.136	0.125	0.045
	1	DCAP_Receivables_Login	0.173	0.24	0.356	0.227	0.07
	1	#Procurement_Click_Categories	0.128	0.267	0.592	0.2	0.169
	1	CRM_Manage_Role_Click_CRM_HTML_Admnistr	0.173	0.488	1.12	0.456	0.381
	1	CreateInvoice_click_Payables_Vision_Operations	0.26	0.663	1.276	0.58	0.375
	1	#Procurement_Click_Ergonomic_Supplies	0.127	0.314	0.661	0.333	0.217
	1	DCAP_Receivables_Vision_Operation	0.262	0.569	0.975	0.797	0.27
	1	#Procurement_Click_AnkleSupport	0.258	0.863	2.013	0.924	0.717
	1	CreateInvoice_click_Entry_Invoices	4.72	9.891	14.318	14.299	4.459
	1	CreateInvoice_select_Supplier_Name	0.068	0.134	0.304	0.092	0.098
	1	CreateInvoice_Click_Distributions	0.479	0.706	0.947	0.826	0.188
	1	CreateInvoice_select_taxcode	0.259	0.39	0.762	0.269	0.215
	1	CreateInvoice_select_account	0.477	0.864	1.297	1.094	0.341
	1	CRM_Manage_Role_Click_Setup_Home	0.159	0.786	1.513	1.298	0.624
	1	CreateInvoice_select_Save	1.419	1.593	1.779	1.755	0.174
	1	CreateInvoice_logout	0.056	0.167	0.286	0.241	0.099
	1	#Procurement_Click_AddToCart	2.984	4.895	5.925	5.598	1.14
	1	CRM_Manage_Role_Click_Users	0.056	0.365	0.561	0.549	0.25
	1	DCAP_Receivables_Click_Transactions	5.178	16.068	28.677	24.992	10.645
	1	#Procurement_Click_ViewCart	0.063	0.187	0.331	0.259	0.111
	1	DCAP_Receivables_ShipTo	0.773	0.947	1.249	0.883	0.18
	1	DCAP_Receivables_EnterTransDetails_Save	0.432	0.518	0.651	0.54	0.087
	1	DCAP_Receivables_Click_LineItems_EnterDet	1.491	1.713	1.88	1.741	0.14
	1	DCAP_Receivables_LineItems_Save	0.34	0.368	0.442	0.349	0.043
	1	DCAP_Receivables_Logout	0.056	0.116	0.272	0.069	0.09
	1	CRM_Manage_Role_Click_User_Maintenance	0.056	0.293	0.652	0.397	0.249
	1	#Procurement_Delete_Item	0.068	0.09	0.143	0.079	0.031
	1	#Procurement_Logout	0.069	0.089	0.137	0.08	0.028
	1	CRM_Manage_Role_Search	0.078	0.098	0.143	0.093	0.026
	1	CRM_Manage_Role_Select_Test_user	0.068	0.082	0.115	0.078	0.019
	1	CRM_Manage_Role_Select_Role	0.084	0.121	0.177	0.141	0.039
	1	CRM_Manage_Role_Click_Update	0.099	0.107	0.123	0.103	0.009
	1	CRM_Manage_Role_Logout	0.098	0.114	0.159	0.098	0.026

168340

Expected Results

- 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.

Results

Oracle Apps Traffic from Branch 1 to DCa without WAAS passed.

Oracle Apps 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 CSM 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 17ms of latency. The response times were then measured and performance results were quantified.During this test WAAS is disabled.

Test Procedure

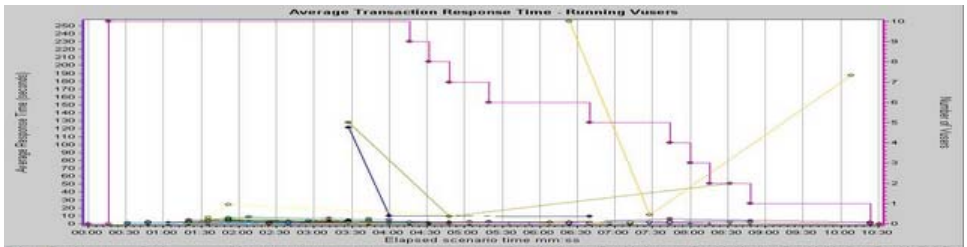
The procedure used to perform the Oracle Apps Traffic from Branch 2 to DCa without WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 3 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 4 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The 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 determine the percentage of transactions that met predefined response times.



Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev.
	Average Transac	1	Create_Project_forms_Clck_Apps_Logon_Links	0	0.019	0.052	0	0.025
	Average Transac	1	Create_Project_forms_Clck_Copy_To	0.203	0.497	0.695	0.564	0.208
	Average Transac	1	Create_Project_forms_Clck_find	0.547	0.959	1.345	0.684	0.349
	Average Transac	1	Create_Project_forms_clck_open	0.216	0.647	1.669	0.657	0.609
	Average Transac	1	Create_Project_forms_Clck_project_link	3.312	47.055	121.699	10.154	52.703
	Average Transac	1	Create_Project_forms_Clck_project_name	0.597	1.07	1.343	1.27	0.336
	Average Transac	1	Create_Project_forms_Clck_project_vision_service	1.116	3.02	4.053	3.091	1.526
	Average Transac	1	Create_Project_forms_EBusiness_Links	0.206	1.646	3.313	1.418	1.279
	Average Transac	1	Create_Project_forms_Login	1.036	1.415	2.01	1.199	0.426
	Average Transac	1	Create_Project_forms_Logout	0.134	0.573	0.803	0.776	0.309
	Average Transac	1	Create_Project_forms_select_Change_Status	0.32	0.831	1.189	1.065	0.362
	Average Transac	1	Create_Project_forms_select_Customer_name	0.598	0.979	1.361	1.069	0.354
	Average Transac	1	Create_Project_forms_select_organization	0.547	1.097	1.417	1.326	0.39
	Average Transac	1	Create_Project_forms_select_PM	0.724	1.037	1.225	1.162	0.223
	Average Transac	1	CreateInvoice_Clck_Apps_Logon_Links	0	0.065	0.188	0.07	0.077
	Average Transac	1	CreateInvoice_Clck_Distributions	1.382	1.584	1.712	1.657	0.144
	Average Transac	1	CreateInvoice_Clck_EBusiness_Links	0.637	1.724	4.731	0.818	1.737
	Average Transac	1	CreateInvoice_clck_Entry_Invoices	9.322	62.607	127.392	51.108	49.863
	Average Transac	1	CreateInvoice_clck_Payables_Vision_Operations	1.658	2.252	2.95	2.148	0.533
	Average Transac	1	CreateInvoice_Login	0.819	1.112	1.525	0.993	0.301
	Average Transac	1	CreateInvoice_logout	0.6	0.769	0.987	0.716	0.162
	Average Transac	1	CreateInvoice_select_account	2.145	2.321	2.61	2.209	0.206
	Average Transac	1	CreateInvoice_select_Save	2.662	3.092	3.694	2.891	0.442
	Average Transac	1	CreateInvoice_select_Supplier_Name	0.485	0.546	0.595	0.567	0.044
	Average Transac	1	CreateInvoice_select_taxcode	1.309	1.432	1.529	1.456	0.092
	Average Transac	1	CRM_Manage_Role_Clck_Apps_Logon_Links	0	0.023	0.066	0.025	0.027
	Average Transac	1	CRM_Manage_Role_Clck_CRM_HTML_Administ	0.977	1.096	1.315	1.096	0.179
	Average Transac	1	CRM_Manage_Role_Clck_Setup_Home	0.729	1.626	3.921	0.777	1.517
	Average Transac	1	CRM_Manage_Role_Clck_Update	0.327	1.667	2.986	1.983	0.965
	Average Transac	1	CRM_Manage_Role_Clck_User_Maintenance	0.691	0.719	0.791	0.684	0.051
	Average Transac	1	CRM_Manage_Role_Clck_Users	0.522	1.154	1.429	1.399	0.37
	Average Transac	1	CRM_Manage_Role_Business_Links	0.571	0.891	1.29	0.96	0.269
	Average Transac	1	CRM_Manage_Role_Login	0.74	0.873	1.049	0.83	0.13
	Average Transac	1	CRM_Manage_Role_Logout	0.433	0.597	0.87	0.593	0.168
	Average Transac	1	CRM_Manage_Role_Search	0.286	1.97	3.036	2.918	1.116
	Average Transac	1	CRM_Manage_Role_Select_Role	0.279	1.355	2.983	0.802	1.171
	Average Transac	1	CRM_Manage_Role_Select_Test_user	0.227	0.98	2.479	0.235	1.059
	Average Transac	1	DCAP_Receivables_Clck_LineItems_EnterDet	1.851	3.325	6.038	2.086	1.921
	Average Transac	1	DCAP_Receivables_Clck_Transactions	11.321	151.23	255.4	186.968	102.799
	Average Transac	1	DCAP_Receivables_EBusiness_Links	0.071	0.442	0.752	0.751	0.312
	Average Transac	1	DCAP_Receivables_EnterTransDetails_Save	0.42	0.476	0.586	0.421	0.078
	Average Transac	1	DCAP_Receivables_LineItems_Save	0.198	0.251	0.344	0.21	0.066
	Average Transac	1	DCAP_Receivables_Login	0.342	1.219	2.376	1.19	0.737
	Average Transac	1	DCAP_Receivables_LogonLinks	0	0.102	0.22	0.187	0.102
	Average Transac	1	DCAP_Receivables_Logout	0.162	0.895	2.071	0.182	0.895
	Average Transac	1	DCAP_Receivables_ShipTo	0.795	1.144	1.646	1.002	0.366
	Average Transac	1	DCAP_Receivables_Vision_Operation	1.502	11.316	24.254	9.194	9.547
	Average Transac	1	#Procurement_AppsLogonLinks	0	0.044	0.098	0.034	0.041
	Average Transac	1	#Procurement_Choose	1.216	3.012	6.61	2.794	2.163
	Average Transac	1	#Procurement_Clck_AddToCart	4.997	6.496	8.588	6.337	1.368
	Average Transac	1	#Procurement_Clck_AnkleSuppot	3.529	5.04	7.365	4.698	1.417
	Average Transac	1	#Procurement_Clck_Categories	0.392	1.707	3.574	2.399	1.345
	Average Transac	1	#Procurement_Clck_Ergonomic_Supplies	1.335	2.094	2.987	2.04	0.643
	Average Transac	1	#Procurement_Clck_ViewCart	0.379	1.179	1.787	1.602	0.557
	Average Transac	1	#Procurement_Delete_Item	0.563	0.641	0.718	0.719	0.077
	Average Transac	1	#Procurement_EbusinessHomepage	0.779	1.17	1.935	0.797	0.541
	Average Transac	1	#Procurement_Homepage	0	0.075	0.129	0.096	0.054
	Average Transac	1	#Procurement_Logon	0.999	2.026	3.715	1.987	1.037
	Average Transac	1	#Procurement_Logout	0.526	0.727	1.076	0.58	0.248
	RunningVusers	1	Run	0	4.231	10	4	3.423

168341

Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	#Procurement_Homepage	0	0.056	0.129	0.096	0.057
	1	#Procurement_AppsLogonLinks	0	0.033	0.098	0.034	0.04
	1	#Procurement_EbusinessHomepage	0.227	1.361	3.642	0.797	1.337
	1	#Procurement_Logon	0.999	2.026	3.715	1.987	1.037
	1	CRM_Manage_Role_Clck_Apps_Logon_Links	0	0.023	0.066	0.025	0.027
	1	CRM_Manage_Role_EBusiness_Links	0.571	0.891	1.29	0.96	0.269
	1	CRM_Manage_Role_Login	0.702	0.84	1.049	0.83	0.129
	1	DCAP_Receivables_LogonLinks	0	0.102	0.22	0.187	0.102
	1	DCAP_Receivables_EBusiness_Links	0.071	0.443	0.752	0.752	0.313
	1	DCAP_Receivables_Login	0.342	1.219	2.376	1.19	0.737
	1	#Procurement_Choose	1.216	3.012	6.61	2.794	2.163
	1	CreateInvoice_Clck_Apps_Logon_Links	0	0.065	0.188	0.07	0.077
	1	CreateInvoice_Clck_EBusiness_Links	0.637	1.724	4.731	0.818	1.737
	1	CreateInvoice_Login	0.819	1.116	1.525	0.993	0.302
	1	CRM_Manage_Role_Clck_CRM_HTML_Administ	0.834	1.041	1.315	1.096	0.184
	1	#Procurement_Clck_Categories	0.392	1.707	3.574	2.399	1.345
	1	Create_Project_forms_Clck_Apps_Logon_Links	0	0.026	0.063	0.042	0.027
	1	CreateInvoice_clck_Payables_Vision_Operations	1.658	2.259	2.95	2.148	0.534
	1	Create_Project_forms_EBusiness_Links	0.206	2.062	5.48	1.418	2.024
	1	Create_Project_forms_Login	1.036	1.564	2.517	1.593	0.576
	1	#Procurement_Clck_Ergonomic_Supplies	1.335	2.089	2.907	2.682	0.719
	1	Create_Project_forms_Clck_project_vision_service	1.116	3.478	6.017	3.69	1.748
	1	DCAP_Receivables_Vision_Operation	1.028	8.863	24.254	8.194	9.302
	1	#Procurement_Clck_AnkleSuppot	3.529	5.04	7.365	4.698	1.417
	1	CRM_Manage_Role_Clck_Setup_Home	0.729	2.362	3.996	3.946	1.609
	1	#Procurement_Clck_AddToCart	4.997	6.496	8.588	6.337	1.368
	1	CRM_Manage_Role_Clck_Users	0.522	1.154	1.429	1.399	0.37
	1	#Procurement_Clck_ViewCart	0.379	1.179	1.787	1.602	0.557
	1	CRM_Manage_Role_Clck_User_Maintenance	0.657	0.737	0.925	0.684	0.109
	1	#Procurement_Delete_Item	0.369	0.641	0.943	0.758	0.224
	1	#Procurement_Logout	0.526	0.814	1.519	0.633	0.409
	1	CRM_Manage_Role_Search	0.286	1.97	3.036	2.918	1.116
	1	CRM_Manage_Role_Select_Test_user	0.227	1.354	3.109	1.847	1.209
	1	CRM_Manage_Role_Select_Role	0.279	1.762	3.9	2.066	1.395
	1	CreateInvoice_clck_Entry_Invoices	9.322	63.255	127.392	51.108	49.063
	1	Create_Project_forms_Clck_project_link	3.312	47.802	121.699	10.154	53.041
	1	Create_Project_forms_Clck_project_name	0.597	1.073	1.343	1.27	0.335
	1	Create_Project_forms_Clck_find	0.547	0.964	1.345	0.684	0.35
	1	Create_Project_forms_Clck_Copy_To	0.203	0.489	0.695	0.564	0.208
	1	CreateInvoice_select_Supplier_Name	0.485	0.546	0.595	0.567	0.044
	1	Create_Project_forms_select_PM	0.724	1.039	1.225	1.162	0.222
	1	Create_Project_forms_select_Customer_name	0.598	0.983	1.361	1.069	0.354
	1	Create_Project_forms_select_organization	0.547	1.1	1.417	1.326	0.39
	1	CreateInvoice_Clck_Distributions	1.382	1.585	1.712	1.657	0.144
	1	Create_Project_forms_clck_open	0.216	0.855	1.669	0.657	0.61
	1	CreateInvoice_select_taxcode	1.309	1.433	1.529	1.456	0.092
	1	Create_Project_forms_select_Change_Status	0.32	0.834	1.189	1.065	0.361
	1	CRM_Manage_Role_Clck_Update	0.327	1.667	2.986	1.983	0.965
	1	CreateInvoice_select_account	2.145	2.324	2.61	2.209	0.207
	1	Create_Project_forms_Logout	0.134	0.573	0.803	0.776	0.309
	1	CreateInvoice_select_Save	2.662	3.088	3.694	2.891	0.444
	1	CreateInvoice_logout	0.6	0.77	0.987	0.716	0.163
	1	CRM_Manage_Role_Logout	0.433	0.597	0.87	0.593	0.168
	1	DCAP_Receivables_Clck_Transactions	11.321	152.271	255.4	186.968	102.988
	1	DCAP_Receivables_ShipTo	0.795	1.149	1.646	1.002	0.367
	1	DCAP_Receivables_EnterTransDetails_Save	0.42	0.477	0.586	0.421	0.076
	1	DCAP_Receivables_Clck_LineItems_EnterDet	1.851	3.352	6.038	2.086	1.93
	1	DCAP_Receivables_LineItems_Save	0.198	0.252	0.344	0.21	0.067
	1	DCAP_Receivables_Logout	0.162	0.918	2.071	0.182	0.9

168342

Expected Results

- We expect Oracle transactions submitted through Load Runner to be completed successfully.
- We expect transaction response times to be slightly higher due to latency and bandwidth contraints and with WAAS being 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.

Results

Oracle Apps Traffic from Branch 2 to DCa without WAAS passed.

Oracle Apps 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 CSM 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 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 Apps Traffic from Branch 3 to DCa without WAAS test follows:

Step 1 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.

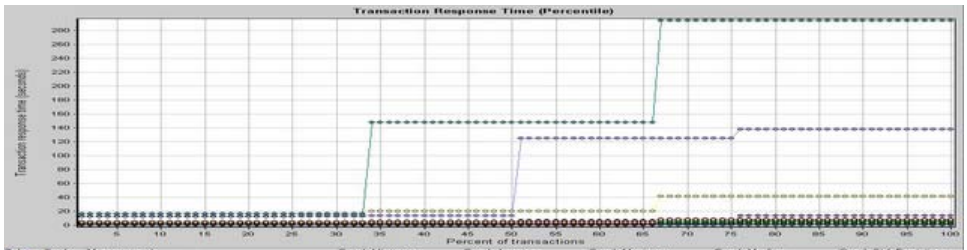
Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 3 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 4 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps determine the percentage of transactions that met predefined response times.

Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev.
	Average Transac	1	Create_Project_forms_Clck_Apps_Logon_Links	0	0.039	0.076	0.076	0.039
	Average Transac	1	Create_Project_forms_Clck_Copy_To	0.518	0.690	0.922	0.655	0.168
	Average Transac	1	Create_Project_forms_Clck_find	0.599	0.826	0.954	0.925	0.161
	Average Transac	1	Create_Project_forms_clck_open	0.529	1.341	1.936	1.655	0.579
	Average Transac	1	Create_Project_forms_Clck_project_link	12.802	24.653	41.326	19.832	12.134
	Average Transac	1	Create_Project_forms_Clck_project_name	0.987	1.327	1.978	1.016	0.46
	Average Transac	1	Create_Project_forms_Clck_project_vision_service	1.168	3.274	6.035	3.449	1.787
	Average Transac	1	Create_Project_forms_EBusiness_Links	0.352	0.876	1.204	1.168	0.345
	Average Transac	1	Create_Project_forms_Login	1.229	2.931	5.224	3.6	1.596
	Average Transac	1	Create_Project_forms_Logout	0.627	0.912	1.181	1.027	0.279
	Average Transac	1	Create_Project_forms_select_Change_Status	0.783	1.243	1.496	1.448	0.325
	Average Transac	1	Create_Project_forms_select_Customer_name	0.984	1.283	1.763	1.092	0.342
	Average Transac	1	Create_Project_forms_select_organization	0.984	1.334	1.804	1.215	0.345
	Average Transac	1	Create_Project_forms_select_PM	0.986	1.329	1.629	1.172	0.362
	Average Transac	1	Createinvoice_Clck_Apps_Logon_Links	0	0.104	0.207	0.207	0.104
	Average Transac	1	Createinvoice_Clck_Distributions	2.455	2.639	3.035	2.604	0.246
	Average Transac	1	Createinvoice_Clck_EBusiness_Links	0.927	1.433	1.861	1.512	0.386
	Average Transac	1	Createinvoice_clck_Entry_Invoices	12.668	91.738	137.804	124.741	56.164
	Average Transac	1	Createinvoice_clck_Payables_Vision_Operations	2.906	4.929	6.654	6.288	1.954
	Average Transac	1	Createinvoice_Login	1.12	2.304	3.604	3.329	1.167
	Average Transac	1	Createinvoice_logout	1.056	1.249	1.634	1.057	0.272
	Average Transac	1	Createinvoice_select_account	2.400	2.662	2.86	2.717	0.189
	Average Transac	1	Createinvoice_select_Save	3.316	3.956	4.928	3.974	0.688
	Average Transac	1	Createinvoice_select_Supplier_Name	0.627	0.695	0.801	0.657	0.076
	Average Transac	1	Createinvoice_select_taxcode	1.71	1.88	2.126	1.803	0.179
	Average Transac	1	CRM_Manage_Role_Clck_Apps_Logon_Links	0	0.037	0.11	0	0.052
	Average Transac	1	CRM_Manage_Role_Clck_CRM_HTML_Administ	2.056	3.577	5.444	3.231	1.404
	Average Transac	1	CRM_Manage_Role_Clck_Setup_Home	1.488	2.753	4.841	2.767	1.298
	Average Transac	1	CRM_Manage_Role_Clck_Update	1.356	1.754	2.133	1.774	0.310
	Average Transac	1	CRM_Manage_Role_Clck_User_Maintenance	0.564	1.869	4.797	1.146	1.703
	Average Transac	1	CRM_Manage_Role_Clck_Users	1.939	2.33	2.671	2.478	0.274
	Average Transac	1	CRM_Manage_Role_EBusiness_Links	0.488	0.711	1.151	0.494	0.311
	Average Transac	1	CRM_Manage_Role_Login	1.191	1.332	1.436	1.369	0.103
	Average Transac	1	CRM_Manage_Role_Logout	0.952	1.042	1.121	1.067	0.071
	Average Transac	1	CRM_Manage_Role_Search	0.678	1.912	2.552	2.295	0.729
	Average Transac	1	CRM_Manage_Role_Select_Role	1.238	2.032	3.744	1.824	1.014
	Average Transac	1	CRM_Manage_Role_Select_Test_user	0.585	1.57	2.159	1.973	0.608
	Average Transac	1	DCAP_Receivables_Clck_LineItems_EnterDet	4.034	5.329	7.822	4.133	1.763
	Average Transac	1	DCAP_Receivables_Clck_Transactions	16.244	152.743	294.173	147.913	113.516
	Average Transac	1	DCAP_Receivables_EBusiness_Links	0.222	1.524	3.512	0.938	1.428
	Average Transac	1	DCAP_Receivables_EnterTransDetails_Save	0.476	0.476	0.479	0.479	0.002
	Average Transac	1	DCAP_Receivables_LineItems_Save	0.311	0.329	0.353	0.323	0.018
	Average Transac	1	DCAP_Receivables_Login	0.6	2.419	3.618	3.061	1.136
	Average Transac	1	DCAP_Receivables_LogonLinks	0	0.476	1.427	0.001	0.672
	Average Transac	1	DCAP_Receivables_Logout	0.522	1.024	2.021	0.528	0.705
	Average Transac	1	DCAP_Receivables_ShipTo	1.491	1.79	2.275	1.605	0.346
	Average Transac	1	DCAP_Receivables_Vision_Operation	1.626	4.587	7.408	6.683	2.497
	Average Transac	1	#Procurement_AppsLogonLinks	0	0.109	0.287	0.147	0.119
	Average Transac	1	#Procurement_Choose	2.287	5.548	13.633	3.503	4.688
	Average Transac	1	#Procurement_Clck_AddToCart	4.877	6.355	8.219	5.971	1.391
	Average Transac	1	#Procurement_Clck_AnkleSuppot	2.424	3.76	5.677	3.18	1.39
	Average Transac	1	#Procurement_Clck_Categories	1.681	3.022	5.550	2.577	1.499
	Average Transac	1	#Procurement_Clck_Ergonomic_Supplies	1.039	1.935	2.697	2.372	0.645
	Average Transac	1	#Procurement_Clck_ViewCart	0.329	0.844	1.51	0.652	0.494
	Average Transac	1	#Procurement_Delete_Item	0.265	1.048	2.323	0.555	0.969
	Average Transac	1	#Procurement_EbusinessHomepage	0.969	1.937	3.86	1.472	1.183
	Average Transac	1	#Procurement_Homepage	0	0.542	1.768	0.401	0.726
	Average Transac	1	#Procurement_Logon	1.478	3.13	7.842	1.708	2.722
	Average Transac	1	#Procurement_Logout	0.479	1.211	2.187	1.046	0.675
	RunningVusers	1	Run	0	4	10	4	3.464

168343



Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	Create_Project_forms_Clck_Apps_Logon_Links	0	0.039	0.076	0.076	0.039
	1	Create_Project_forms_EBusiness_Links	0.352	0.876	1.204	1.168	0.345
	1	Create_Project_forms_Login	1.229	2.931	5.224	3.6	1.596
	1	Create_Project_forms_Clck_project_vision_service	1.168	3.274	6.035	3.449	1.787
	1	DCAP_Receivables_LogonLinks	0.001	0.714	2.004	0.95	0.622
	1	DCAP_Receivables_EBusiness_Links	0.222	2.021	3.798	3.226	1.52
	1	DCAP_Receivables_Login	0.6	2.419	3.618	3.061	1.136
	1	#Procurement_Homepage	0	0.542	1.768	0.401	0.726
	1	CRM_Manage_Role_Clck_Apps_Logon_Links	0	0.095	0.145	0.076	0.06
	1	CRM_Manage_Role_EBusiness_Links	0.488	0.823	1.155	1.155	0.332
	1	CRM_Manage_Role_Login	1.131	1.341	1.606	1.436	0.191
	1	Create_Project_forms_Clck_project_link	12.802	24.82	41.326	19.832	12.186
	1	Create_Project_forms_Clck_project_name	0.987	1.333	1.978	1.016	0.463
	1	Create_Project_forms_Clck_find	0.599	0.827	0.954	0.925	0.161
	1	#Procurement_AppsLogonLinks	0	0.109	0.287	0.147	0.119
	1	Create_Project_forms_Clck_Copy_To	0.518	0.701	0.922	0.655	0.168
	1	#Procurement_EbusinessHomepage	0.969	1.837	3.86	1.472	1.183
	1	Create_Project_forms_select_PM	0.986	1.334	1.029	1.172	0.363
	1	Create_Project_forms_select_Customer_name	0.984	1.288	1.763	1.092	0.343
	1	#Procurement_Logon	1.478	3.13	7.842	1.708	2.722
	1	Createinvoice_Clck_Appc_Logon_Links	0	0.104	0.223	0.192	0.104
	1	Createinvoice_Clck_EBusiness_Links	0.927	1.54	2.43	1.512	0.555
	1	Createinvoice_Login	1.12	2.304	3.604	3.329	1.167
	1	Create_Project_forms_select_organization	0.984	1.339	1.804	1.215	0.347
	1	Create_Project_forms_clck_open	0.529	1.346	1.938	1.655	0.579
	1	Create_Project_forms_select_Change_Status	0.783	1.245	1.496	1.448	0.325
	1	CRM_Manage_Role_Clck_CRM_HTML_Administ	2.056	3.48	5.444	3.28	1.226
	1	Create_Project_forms_Logout	0.627	0.914	1.181	1.027	0.279
	1	DCAP_Receivables_Vision_Operation	1.626	4.587	7.408	6.683	2.497
	1	#Procurement_Choose	2.287	5.548	13.633	3.503	4.688
	1	#Procurement_Clck_Categories	1.681	3.022	5.550	2.577	1.499
	1	CRM_Manage_Role_Clck_Setup_Home	1.488	2.753	4.841	2.767	1.298
	1	Createinvoice_clck_Payables_Vision_Operations	2.906	4.929	6.654	6.288	1.954
	1	CRM_Manage_Role_Clck_Users	1.939	2.33	2.671	2.478	0.274
	1	#Procurement_Clck_Ergonomic_Supplies	1.039	1.935	2.697	2.372	0.645
	1	CRM_Manage_Role_Clck_User_Maintenance	0.564	1.869	4.797	1.146	1.703
	1	#Procurement_Clck_AnkleSuppot	2.424	3.78	5.677	3.18	1.396
	1	#Procurement_Clck_AddToCart	4.877	6.374	8.219	5.971	1.396
	1	CRM_Manage_Role_Search	0.678	1.912	2.552	2.295	0.729
	1	#Procurement_Clck_ViewCart	0.329	0.85	1.51	0.652	0.496
	1	CRM_Manage_Role_Select_Test_user	0.585	1.57	2.159	1.973	0.608
	1	#Procurement_Delete_Item	0.265	1.06	2.323	0.555	0.914
	1	#Procurement_Logout	0.479	1.219	2.107	1.046	0.678
	1	CRM_Manage_Role_Select_Role	1.238	2.032	3.744	1.824	1.014
	1	CRM_Manage_Role_Clck_Update	1.356	1.849	2.576	1.774	0.449
	1	Createinvoice_clck_Entry_Invoices	11.863	71.97	137.804	124.741	59.485
	1	Createinvoice_select_Supplier_Name	0.542	0.722	1.061	0.657	0.2
	1	Createinvoice_Clck_Distributions	2.162	2.475	3.047	3.035	0.361
	1	Createinvoice_select_taxcode	1.71	1.942	2.24	2.013	0.204
	1	Createinvoice_select_account	1.878	2.598	2.937	2.86	0.423
	1	Createinvoice_select_Save	3.316	3.956	4.928	3.974	0.608
	1	Createinvoice_logout	1.057	1.345	2.099	1.169	0.437
	1	CRM_Manage_Role_Logout	0.903	1.023	1.121	1.067	0.081
	1	DCAP_Receivables_Clck_Transactions	16.244	154.159	294.173	147.913	113.822
	1	DCAP_Receivables_ShipTo	1.491	1.795	2.275	1.605	0.347
	1	DCAP_Receivables_EnterTransDetails_Save	0.479	0.479	0.479	0.479	0
	1	DCAP_Receivables_Clck_LineItems_EnterDet	4.034	5.354	7.822	4.133	1.772
	1	DCAP_Receivables_LineItems_Save	0.311	0.329	0.353	0.323	0.018
	1	DCAP_Receivables_Logout	0.528	1.036	2.021	0.528	0.707

168344

Expected Results

- We expect Oracle transactions submitted through Load Runner to 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 time out due to high latency.
- 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.

Results

Oracle Apps Traffic from Branch 3 to DCa without WAAS passed.

E-Biz Branches to DCa with WAAS

E-Biz Branches to DCa with WAAS verifies configuration and functionality of Oracle E-Business Application deployment in Data Center A. These tests also validate integration of the Application with GSS and CSM. Application traffic is simulated using Mercury Load runner from all the 3 branch clients. GSS is configured to be authoritative for the wwwin-oefin domain which resolved client requests to the VIP physically located in DCa.

WAAS is Enabled to observe the Application traffic optimization.

The following tests were performed:

- Oracle Apps Traffic from Branch 1 to DCa with WAAS, page 8-28
- Oracle Apps Traffic from Branch 2 to DCa with WAAS, page 8-30
- Oracle Apps Traffic from Branch 3 to DCa with WAAS, page 8-32

Oracle Apps 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 CSM load balanced the connections across the application hosts as expected.

The connection from branch to data center was simulated to have T3 bandwidth and 4 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 Apps Traffic from Branch 1 to DCa with WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

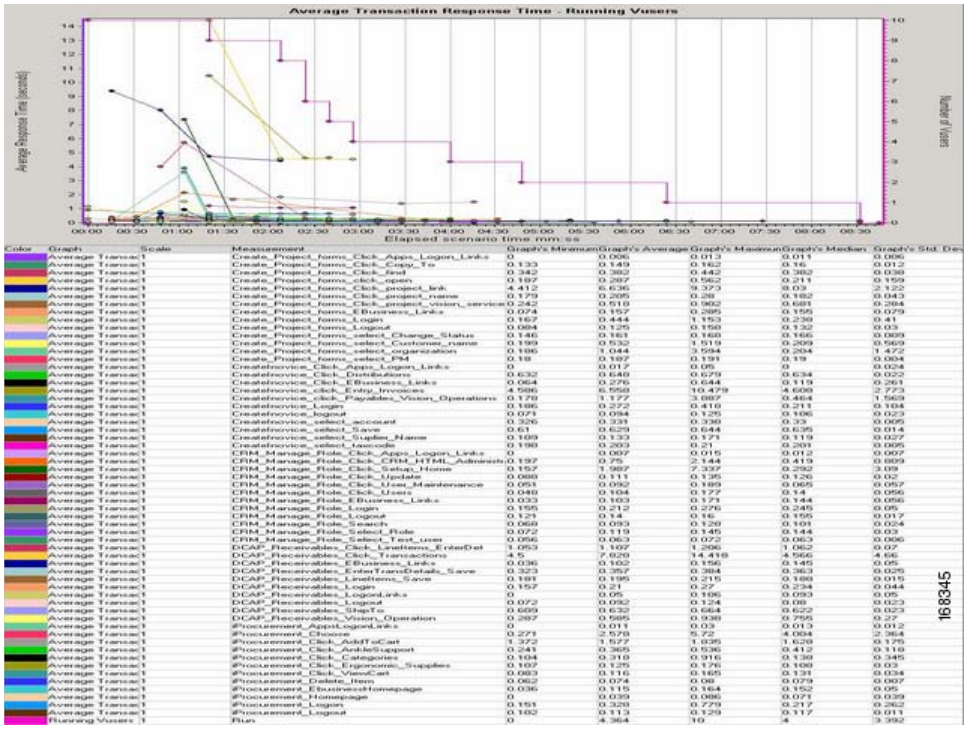
Step 3 Clear the appropriate counters on the WAAS devices by issuing the **clear statistics all** command.

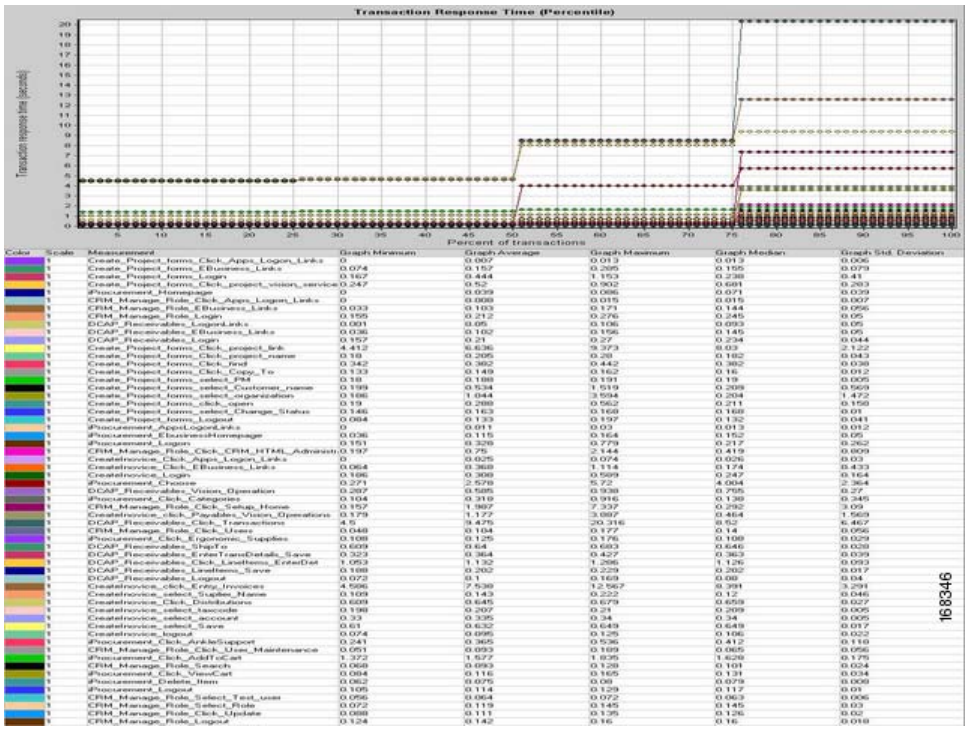
Step 4 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 5 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 6 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays 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.





Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.007	0.013	0.006	
	1	Create_Project_forms_EBusiness_Links	0.074	0.157	0.205	0.155	0.079
	1	Create_Project_forms_Login	0.167	0.444	1.153	0.238	0.41
	1	Create_Project_forms_Click_project_vision_service	0.247	0.52	0.902	0.691	0.283
	1	#Procurement_Homepage	0	0.039	0.086	0.071	0.029
	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.008	0.015	0.015	0.007
	1	CRM_Manage_Role_EBusiness_Links	0.033	0.103	0.171	0.144	0.056
	1	CRM_Manage_Role_Login	0.155	0.212	0.276	0.245	0.05
	1	DCAP_Receivables_LogonLinks	0.001	0.05	0.106	0.093	0.05
	1	DCAP_Receivables_EBusiness_Links	0.036	0.102	0.156	0.145	0.05
	1	DCAP_Receivables_Login	0.157	0.21	0.27	0.234	0.044
	1	Create_Project_forms_Click_project_link	4.412	6.636	9.373	8.03	2.122
	1	Create_Project_forms_Click_project_name	0.18	0.205	0.28	0.182	0.043
	1	Create_Project_forms_Click_find	0.342	0.392	0.442	0.392	0.038
	1	Create_Project_forms_Click_Copy_To	0.133	0.149	0.162	0.16	0.012
	1	Create_Project_forms_select_PM	0.18	0.188	0.191	0.19	0.005
	1	Create_Project_forms_select_Customer_name	0.199	0.534	1.519	0.209	0.569
	1	Create_Project_forms_select_organization	0.186	1.044	3.594	0.204	1.472
	1	Create_Project_forms_click_open	0.19	0.288	0.562	0.211	0.159
	1	Create_Project_forms_select_Change_Status	0.146	0.163	0.168	0.168	0.01
	1	Create_Project_forms_Logout	0.084	0.133	0.197	0.132	0.041
	1	#Procurement_AppsLogonLinks	0	0.011	0.03	0.013	0.012
	1	#Procurement_EbusinessHomepage	0.036	0.115	0.164	0.152	0.05
	1	#Procurement_Logon	0.151	0.328	0.779	0.217	0.262
	1	CRM_Manage_Role_Click_CRM_HTML_Admin	0.197	0.75	2.144	0.419	0.809
	1	CreateInvoice_Click_Apps_Logon_Links	0	0.025	0.074	0.026	0.03
	1	CreateInvoice_Click_EBusiness_Links	0.064	0.368	1.114	0.174	0.433
	1	CreateInvoice_Login	0.186	0.308	0.589	0.247	0.164
	1	#Procurement_Choose	0.271	2.578	5.72	4.004	2.364
	1	DCAP_Receivables_Vision_Operation	0.287	0.585	0.938	0.755	0.27
	1	#Procurement_Click_Categories	0.104	0.319	0.916	0.139	0.345
	1	CRM_Manage_Role_Click_Setup_Home	0.157	1.997	7.337	0.292	3.09
	1	CreateInvoice_click_Payables_Vision_Operations	0.179	1.177	3.897	0.464	1.569
	1	DCAP_Receivables_Click_Transactions	4.5	9.475	20.316	8.52	6.467
	1	CRM_Manage_Role_Click_Users	0.048	0.104	0.177	0.14	0.056
	1	#Procurement_Click_Ergonomic_Supplies	0.108	0.125	0.176	0.108	0.029
	1	DCAP_Receivables_ShipTo	0.609	0.64	0.693	0.646	0.028
	1	DCAP_Receivables_EnterTransDetails_Save	0.323	0.364	0.427	0.363	0.039
	1	DCAP_Receivables_Click_LineItems_EnterDet	1.053	1.132	1.286	1.126	0.093
	1	DCAP_Receivables_LineItems_Save	0.188	0.202	0.229	0.202	0.017
	1	DCAP_Receivables_Logout	0.072	0.1	0.169	0.08	0.04
	1	CreateInvoice_click_Entry_Invoices	4.596	7.538	12.567	8.391	3.291
	1	CreateInvoice_select_Supplier_Name	0.109	0.143	0.222	0.12	0.046
	1	CreateInvoice_Click_Distributions	0.609	0.645	0.679	0.659	0.027
	1	CreateInvoice_select_taxcode	0.198	0.207	0.21	0.209	0.005
	1	CreateInvoice_select_account	0.33	0.335	0.34	0.34	0.005
	1	CreateInvoice_select_Save	0.61	0.632	0.649	0.649	0.017
	1	CreateInvoice_logout	0.074	0.095	0.125	0.106	0.022
	1	#Procurement_Click_AddToSupport	0.241	0.365	0.536	0.412	0.119
	1	CRM_Manage_Role_Click_User_Maintenance	0.051	0.093	0.189	0.065	0.056
	1	#Procurement_Click_AddToCart	1.372	1.577	1.835	1.629	0.175
	1	CRM_Manage_Role_Search	0.068	0.093	0.128	0.101	0.024
	1	#Procurement_Click_ViewCart	0.084	0.116	0.165	0.131	0.034
	1	#Procurement_Delete_Item	0.062	0.075	0.08	0.079	0.008
	1	#Procurement_Logout	0.105	0.114	0.129	0.117	0.01
	1	CRM_Manage_Role_Select_Test_user	0.056	0.064	0.072	0.063	0.006
	1	CRM_Manage_Role_Select_Role	0.072	0.119	0.145	0.145	0.03
	1	CRM_Manage_Role_Click_Update	0.088	0.111	0.135	0.126	0.02
	1	CRM_Manage_Role_Logout	0.124	0.142	0.16	0.16	0.018

Step 7 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
```

Expected Results

- 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.

Results

Oracle Apps Traffic from Branch 1 to DCa with WAAS passed.

Oracle Apps 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 CSM 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 Apps Traffic from Branch 2 to DCa with WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

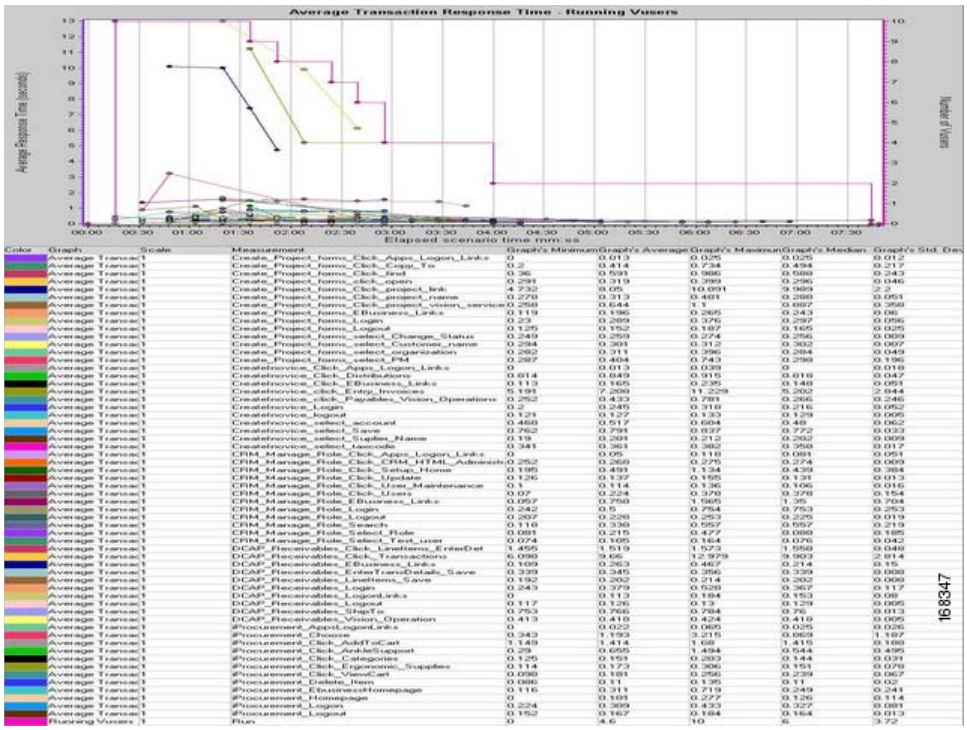
Step 3 Clear the appropriate counters on the WAAS devices by issuing the **clear statistics all** command.

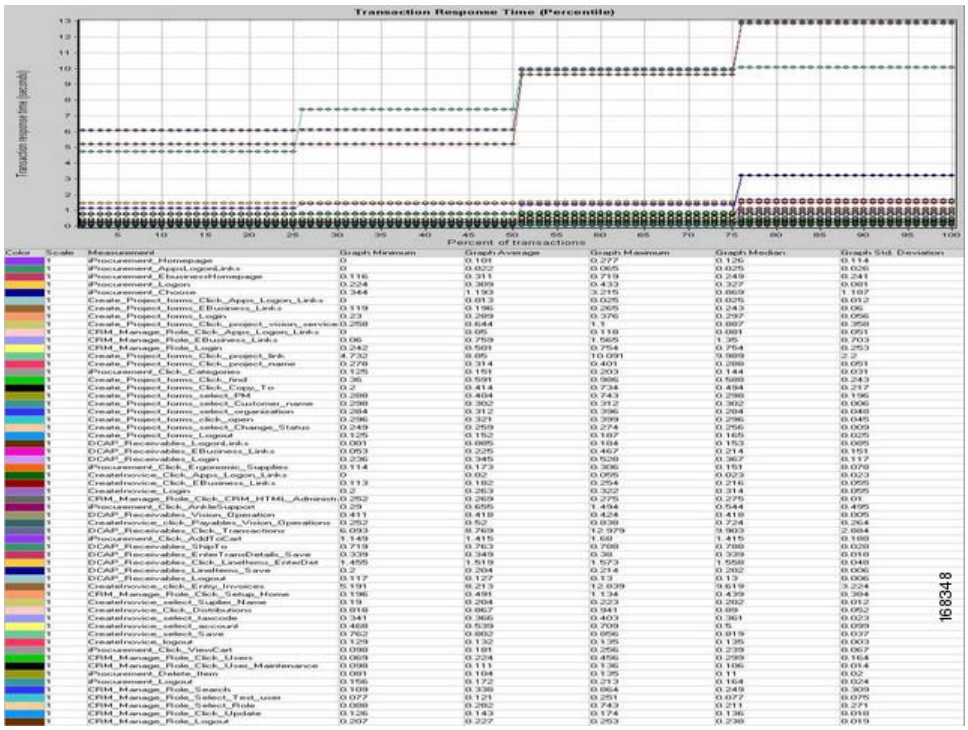
Step 4 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 5 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 6 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps determine the percentage of transactions that met predefined response times.





Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	iProcurement_Homepage	0	0.101	0.277	0.126	0.114
	1	iProcurement_AppsLogonLinks	0	0.022	0.065	0.025	0.026
	1	iProcurement_EbusinessHomepage	0.116	0.311	0.719	0.249	0.241
	1	iProcurement_Logon	0.224	0.309	0.433	0.327	0.081
	1	iProcurement_Choose	0.344	1.193	3.215	0.869	1.187
	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.013	0.025	0.025	0.012
	1	Create_Project_forms_EBusiness_Links	0.119	0.196	0.265	0.243	0.06
	1	Create_Project_forms_Login	0.23	0.289	0.376	0.297	0.056
	1	Create_Project_forms_Click_project_vision_service	0.258	0.644	1.1	0.887	0.358
	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.05	0.118	0.081	0.051
	1	CRM_Manage_Role_EBusiness_Links	0.06	0.759	1.565	1.35	0.703
	1	CRM_Manage_Role_Login	0.242	0.501	0.754	0.754	0.253
	1	Create_Project_forms_Click_project_link	4.732	8.05	10.091	9.989	2.2
	1	Create_Project_forms_Click_project_name	0.278	0.314	0.401	0.288	0.051
	1	iProcurement_Click_Categories	0.125	0.151	0.203	0.144	0.031
	1	Create_Project_forms_Click_find	0.36	0.591	0.986	0.588	0.243
	1	Create_Project_forms_Click_Copy_To	0.2	0.414	0.734	0.494	0.217
	1	Create_Project_forms_select_PM	0.288	0.404	0.743	0.288	0.196
	1	Create_Project_forms_select_Customer_name	0.298	0.302	0.312	0.302	0.006
	1	Create_Project_forms_select_organization	0.284	0.312	0.396	0.284	0.048
	1	Create_Project_forms_click_open	0.296	0.321	0.399	0.296	0.045
	1	Create_Project_forms_select_Change_Status	0.249	0.259	0.274	0.256	0.009
	1	Create_Project_forms_Logout	0.125	0.152	0.187	0.165	0.025
	1	DCAP_Receivables_LogonLinks	0.001	0.085	0.184	0.153	0.085
	1	DCAP_Receivables_EBusiness_Links	0.053	0.225	0.467	0.214	0.151
	1	DCAP_Receivables_Login	0.236	0.345	0.528	0.367	0.117
	1	iProcurement_Click_Ergonomic_Supplies	0.114	0.173	0.306	0.151	0.078
	1	CreateInvoice_Click_Apps_Logon_Links	0	0.02	0.055	0.023	0.023
	1	CreateInvoice_Click_EBusiness_Links	0.113	0.182	0.254	0.216	0.055
	1	CreateInvoice_Login	0.2	0.263	0.322	0.314	0.055
	1	CRM_Manage_Role_Click_CRM_HTML_Admin	0.252	0.269	0.275	0.275	0.01
	1	iProcurement_Click_AddeSupport	0.29	0.655	1.494	0.544	0.495
	1	DCAP_Receivables_Vision_Operation	0.411	0.418	0.424	0.418	0.005
	1	CreateInvoice_click_Payables_Vision_Operations	0.252	0.52	0.938	0.724	0.264
	1	DCAP_Receivables_Click_Transactions	6.093	8.769	12.979	9.903	2.884
	1	iProcurement_Click_AddToCart	1.149	1.415	1.69	1.415	0.188
	1	DCAP_Receivables_SkipTo	0.719	0.763	0.788	0.788	0.028
	1	DCAP_Receivables_EnterTransDetails_Save	0.339	0.349	0.38	0.339	0.018
	1	DCAP_Receivables_Click_LineItems_EnterDet	1.455	1.519	1.573	1.558	0.048
	1	DCAP_Receivables_LineItems_Save	0.2	0.204	0.214	0.202	0.006
	1	DCAP_Receivables_Logout	0.117	0.127	0.13	0.13	0.006
	1	CreateInvoice_click_Entry_Invoices	5.191	8.213	12.939	9.619	3.224
	1	CRM_Manage_Role_Click_Setup_Home	0.196	0.491	1.134	0.439	0.384
	1	CreateInvoice_select_Supplier_Name	0.19	0.204	0.223	0.202	0.012
	1	CreateInvoice_Click_Distributions	0.018	0.067	0.941	0.09	0.052
	1	CreateInvoice_select_taxcode	0.341	0.366	0.403	0.361	0.023
	1	CreateInvoice_select_account	0.468	0.539	0.709	0.5	0.099
	1	CreateInvoice_select_Save	0.762	0.802	0.856	0.819	0.037
	1	CreateInvoice_logout	0.129	0.132	0.135	0.135	0.003
	1	iProcurement_Click_ViewCart	0.098	0.181	0.256	0.239	0.067
	1	CRM_Manage_Role_Click_Users	0.089	0.224	0.456	0.299	0.164
	1	CRM_Manage_Role_Click_User_Maintenance	0.098	0.111	0.136	0.106	0.014
	1	iProcurement_Delete_Item	0.091	0.104	0.135	0.11	0.02
	1	iProcurement_Logout	0.156	0.172	0.213	0.164	0.024
	1	CRM_Manage_Role_Search	0.109	0.336	0.864	0.249	0.309
	1	CRM_Manage_Role_Select_Test_user	0.077	0.121	0.251	0.077	0.075
	1	CRM_Manage_Role_Select_Role	0.088	0.282	0.743	0.211	0.271
	1	CRM_Manage_Role_Click_Update	0.126	0.143	0.174	0.136	0.018
	1	CRM_Manage_Role_Logout	0.207	0.227	0.253	0.238	0.019

Step 7 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
```

Expected Results

- 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.

Results

Oracle Apps Traffic from Branch 2 to DCa with WAAS passed.

Oracle Apps 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 CSM 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 Apps Traffic from Branch 3 to DCa with WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

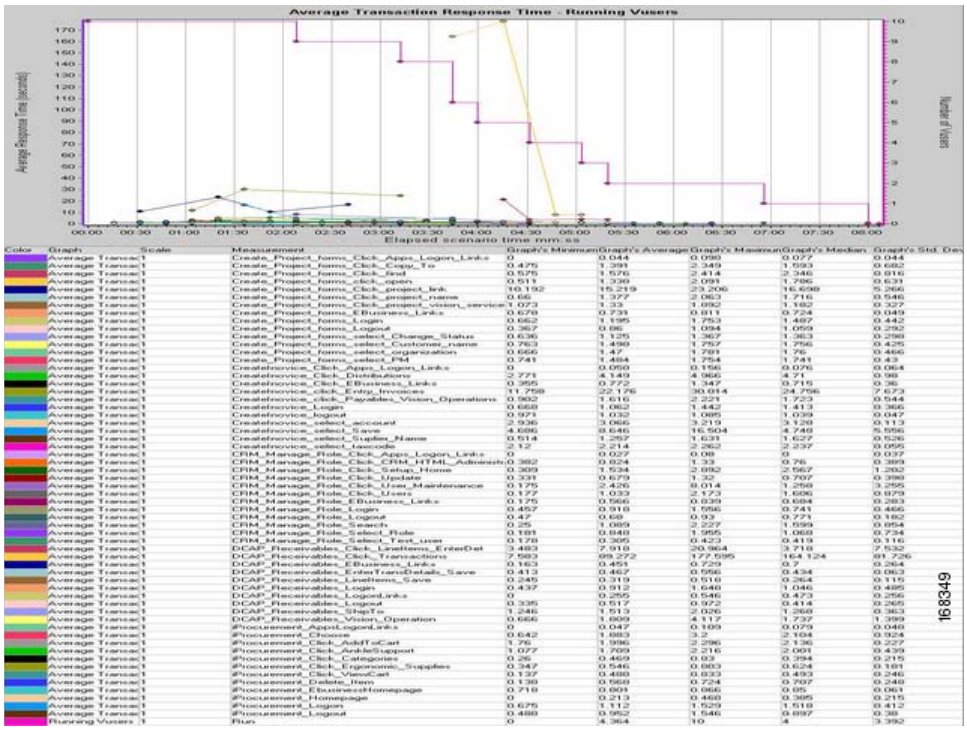
Step 3 Clear the appropriate counters on the WAAS devices by issuing the **clear statistics all** command.

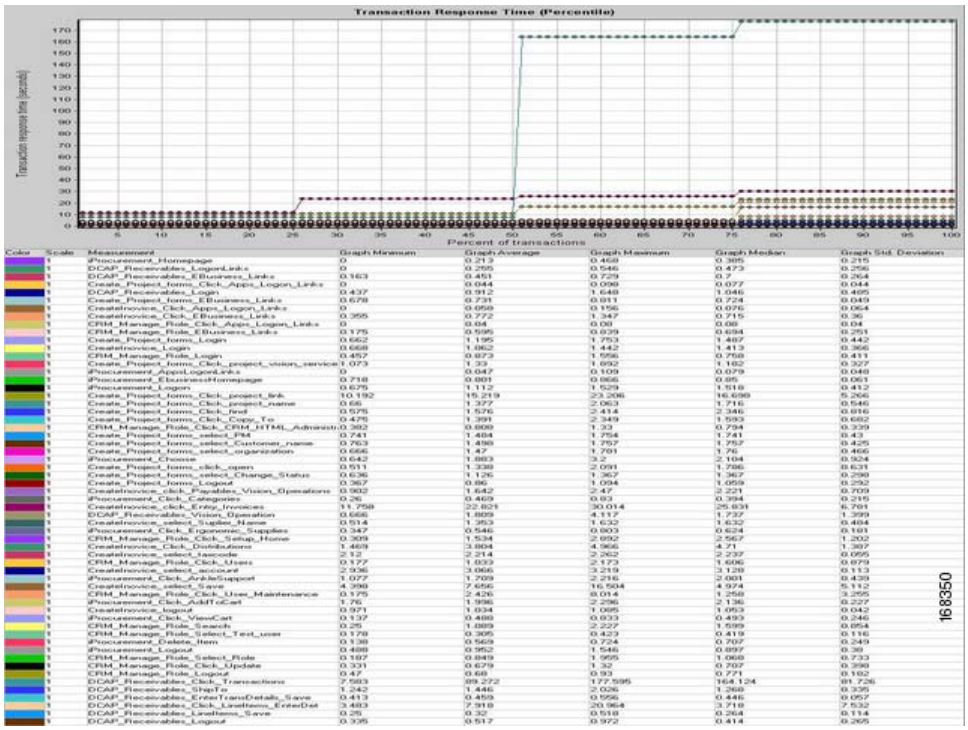
Step 4 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 5 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 6 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps determine the percentage of transactions that met predefined response times.





Step 7 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
```

Expected Results

- 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.

Results

Oracle Apps Traffic from Branch 3 to DCa with WAAS passed.

E-Biz Branches to DCb

E-Biz Branches to DCb verifies configuration and functionality of Oracle E-Business Application deployment in Data Center B. These tests validate integration of the Application with GSS and CSM. Application traffic is simulated using Mercury Load runner from all the 3 branch clients.WAAS is disabled and GSS was configured to be authoritative for the wwwin-oefin domain which resolved client requests to the VIP physically located in DCb.

The following tests were performed:

- Oracle Apps Traffic from Branch 1 to DCb without WAAS, page 8-35
- Oracle Apps Traffic from Branch 2 to DCb without WAAS, page 8-37
- Oracle Apps Traffic from Branch 3 to DCb without WAAS, page 8-39

Oracle Apps 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.

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

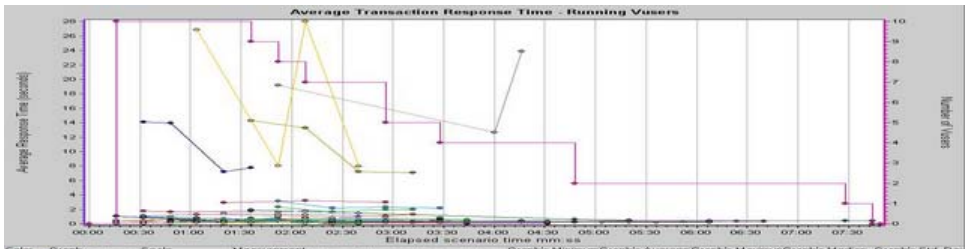
The procedure used to perform the Oracle Apps Traffic from Branch 1 to DCb without WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 3 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 4 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps determine the percentage of transactions that met predefined response times.



Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev
	Average Transact	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.009	0.017	0.016	0.009
	Average Transact	1	Create_Project_forms_Click_Copy_To	0.184	0.234	0.332	0.186	0.069
	Average Transact	1	Create_Project_forms_Click_find	1.639	1.719	1.763	1.751	0.056
	Average Transact	1	Create_Project_forms_click_open	0.264	0.463	0.733	0.543	0.169
	Average Transact	1	Create_Project_forms_Click_project_link	7.24	10.766	14.105	13.365	3.274
	Average Transact	1	Create_Project_forms_Click_project_name	0.375	0.509	0.726	0.425	0.155
	Average Transact	1	Create_Project_forms_Click_project_vision_service	0.329	0.375	0.42	0.417	0.044
	Average Transact	1	Create_Project_forms_EBusiness_Links	0.064	0.154	0.238	0.224	0.078
	Average Transact	1	Create_Project_forms_Login	0.323	0.658	1.069	0.842	0.309
	Average Transact	1	Create_Project_forms_Logout	0.175	0.301	0.459	0.374	0.12
	Average Transact	1	Create_Project_forms_select_Change_Status	0.296	0.517	0.713	0.691	0.186
	Average Transact	1	Create_Project_forms_select_Customer_name	0.437	0.714	1.022	0.683	0.24
	Average Transact	1	Create_Project_forms_select_organization	0.436	0.606	0.866	0.577	0.159
	Average Transact	1	Create_Project_forms_select_PM	0.438	0.64	0.882	0.679	0.151
	Average Transact	1	CreateInvoice_Click_Apps_Logon_Links	0	0.043	0.151	0.022	0.063
	Average Transact	1	CreateInvoice_Click_Distributions	1.633	1.972	2.337	1.959	0.249
	Average Transact	1	CreateInvoice_Click_EBusiness_Links	0.081	0.279	0.689	0.256	0.246
	Average Transact	1	CreateInvoice_click_Entry_Invoices	7.089	10.463	14.269	13.252	3.329
	Average Transact	1	CreateInvoice_click_Payables_Vision_Operations	0.322	0.369	0.434	0.385	0.045
	Average Transact	1	CreateInvoice_Login	0.452	0.72	0.968	0.781	0.186
	Average Transact	1	CreateInvoice_logout	0.165	0.259	0.49	0.202	0.134
	Average Transact	1	CreateInvoice_select_account	0.784	0.955	1.324	0.892	0.216
	Average Transact	1	CreateInvoice_select_Save	2.18	2.445	3.13	2.29	0.397
	Average Transact	1	CreateInvoice_select_Suplier_Name	0.114	0.189	0.387	0.139	0.114
	Average Transact	1	CreateInvoice_select_taxcode	0.289	0.417	0.619	0.397	0.123
	Average Transact	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.011	0.031	0	0.015
	Average Transact	1	CRM_Manage_Role_Click_CRM_HTML_Administ	0.277	0.347	0.43	0.334	0.063
	Average Transact	1	CRM_Manage_Role_Click_Setup_Home	0.496	1.012	1.929	0.611	0.65
	Average Transact	1	CRM_Manage_Role_Click_Update	0.299	0.329	0.365	0.32	0.027
	Average Transact	1	CRM_Manage_Role_Click_User_Maintenance	0.191	0.385	0.741	0.404	0.222
	Average Transact	1	CRM_Manage_Role_Click_Users	0.172	0.287	0.496	0.194	0.140
	Average Transact	1	CRM_Manage_Role_EBusiness_Links	0.040	0.172	0.4	0.069	0.161
	Average Transact	1	CRM_Manage_Role_Login	0.356	0.442	0.687	0.364	0.116
	Average Transact	1	CRM_Manage_Role_Logout	0.366	0.392	0.417	0.404	0.02
	Average Transact	1	CRM_Manage_Role_Search	0.171	0.232	0.32	0.205	0.064
	Average Transact	1	CRM_Manage_Role_Select_Role	0.329	0.357	0.378	0.372	0.02
	Average Transact	1	CRM_Manage_Role_Select_Test_user	0.261	0.266	0.274	0.263	0.005
	Average Transact	1	DCAP_Receivables_Click_LineItems_EnterDet	2.903	3.016	3.176	2.97	0.116
	Average Transact	1	DCAP_Receivables_Click_Transactions	7.973	17.721	28.62	26.891	9.742
	Average Transact	1	DCAP_Receivables_EBusiness_Links	0.043	0.422	1.01	0.59	0.406
	Average Transact	1	DCAP_Receivables_EnterTransDetails_Save	1.088	1.305	1.413	1.412	0.133
	Average Transact	1	DCAP_Receivables_LineItems_Save	0.377	0.421	0.464	0.42	0.036
	Average Transact	1	DCAP_Receivables_Login	0.354	0.628	0.904	0.647	0.249
	Average Transact	1	DCAP_Receivables_LogonLinks	0	0.195	0.474	0.307	0.204
	Average Transact	1	DCAP_Receivables_Logout	0.172	0.236	0.355	0.181	0.084
	Average Transact	1	DCAP_Receivables_ShipTo	1.432	1.566	1.787	1.613	0.149
	Average Transact	1	DCAP_Receivables_Vision_Operation	0.427	0.536	0.716	0.471	0.127
	Average Transact	1	iProcurement_AppsLogonLinks	0	0.07	0.132	0.079	0.054
	Average Transact	1	iProcurement_Choose	1.069	1.2	1.321	1.296	0.109
	Average Transact	1	iProcurement_Click_AddToCart	12.667	18.583	23.895	19.188	4.603
	Average Transact	1	iProcurement_Click_AnkleSupport	0.521	0.631	0.694	0.656	0.065
	Average Transact	1	iProcurement_Click_Categories	0.282	0.434	0.561	0.53	0.115
	Average Transact	1	iProcurement_Click_Ergonomic_Supplies	0.246	0.277	0.303	0.283	0.023
	Average Transact	1	iProcurement_Click_ViewCart	0.274	0.364	0.534	0.367	0.105
	Average Transact	1	iProcurement_Delete_Item	0.18	0.183	0.186	0.186	0.003
	Average Transact	1	iProcurement_EbusinessHomepage	0.078	0.578	0.982	0.675	0.375
	Average Transact	1	iProcurement_Homepage	0	0.056	0.111	0.057	0.045
	Average Transact	1	iProcurement_Login	0.416	0.658	1.031	0.766	0.259
	Average Transact	1	iProcurement_Logout	0.203	0.22	0.237	0.237	0.017
	Running Users	1	Run	0	4.162	10	4	3.664

168351

Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.009	0.017	0.017	0.009
	1	Create_Project_forms_EBusiness_Links	0.064	0.154	0.238	0.224	0.078
	1	Create_Project_forms_Login	0.323	0.658	1.069	0.842	0.309
	1	Create_Project_forms_Click_project_vision_service	0.329	0.376	0.42	0.42	0.045
	1	iProcurement_Homepage	0	0.042	0.111	0.057	0.046
	1	DCAP_Receivables_LogonLinks	0	0.195	0.474	0.307	0.204
	1	DCAP_Receivables_EBusiness_Links	0.046	0.423	1.01	0.59	0.405
	1	DCAP_Receivables_Login	0.354	0.628	0.904	0.647	0.249
	1	iProcurement_AppsLogonLinks	0	0.053	0.132	0.079	0.056
	1	iProcurement_EbusinessHomepage	0.079	0.464	0.982	0.675	0.39
	1	iProcurement_Logon	0.42	0.659	1.031	0.766	0.257
	1	Create_Project_forms_Click_project_link	7.24	10.766	14.105	13.365	3.274
	1	Create_Project_forms_Click_project_name	0.375	0.563	1.095	0.425	0.285
	1	Create_Project_forms_Click_find	1.639	1.729	1.819	1.751	0.066
	1	Create_Project_forms_Click_Copy_To	0.186	0.259	0.462	0.201	0.118
	1	Create_Project_forms_select_PM	0.438	0.65	0.883	0.882	0.195
	1	Create_Project_forms_select_Customer_name	0.437	0.706	1.022	0.859	0.242
	1	Create_Project_forms_select_organization	0.436	0.606	0.866	0.577	0.159
	1	Create_Project_forms_click_open	0.264	0.463	0.733	0.543	0.188
	1	Create_Project_forms_select_Change_Status	0.296	0.517	0.713	0.691	0.186
	1	Create_Project_forms_Logout	0.175	0.301	0.459	0.374	0.12
	1	iProcurement_Choose	1.069	1.2	1.321	1.296	0.109
	1	DCAP_Receivables_Vision_Operation	0.427	0.583	0.994	0.471	0.238
	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.016	0.048	0.015	0.02
	1	CRM_Manage_Role_EBusiness_Links	0.048	0.229	0.623	0.177	0.233
	1	CRM_Manage_Role_Login	0.356	0.484	0.79	0.424	0.179
	1	iProcurement_Click_Categories	0.282	0.434	0.561	0.53	0.115
	1	CRM_Manage_Role_Click_CRM_HTML_Administ	0.277	0.368	0.575	0.334	0.122
	1	iProcurement_Click_Ergonomic_Supplies	0.246	0.279	0.303	0.288	0.021
	1	DCAP_Receivables_Click_Transactions	7.973	17.721	28.62	26.891	9.742
	1	CreateInvoice_Click_Apps_Logon_Links	0	0.043	0.151	0.022	0.063
	1	CreateInvoice_Click_EBusiness_Links	0.081	0.279	0.689	0.256	0.246
	1	CreateInvoice_Login	0.452	0.72	0.968	0.781	0.186
	1	iProcurement_Click_AnkleSupport	0.521	0.631	0.694	0.656	0.065
	1	DCAP_Receivables_ShipTo	1.432	1.566	1.787	1.613	0.149
	1	DCAP_Receivables_EnterTransDetails_Save	1.088	1.305	1.413	1.413	0.133
	1	DCAP_Receivables_Click_LineItems_EnterDet	2.903	3.056	3.274	3.077	0.14
	1	DCAP_Receivables_LineItems_Save	0.377	0.431	0.505	0.423	0.046
	1	DCAP_Receivables_Logout	0.154	0.266	0.557	0.181	0.168
	1	CreateInvoice_click_Payables_Vision_Operations	0.322	0.369	0.434	0.385	0.045
	1	CRM_Manage_Role_Click_Setup_Home	0.496	1.241	1.968	1.891	0.69
	1	CRM_Manage_Role_Click_Users	0.172	0.339	0.724	0.268	0.225
	1	CreateInvoice_click_Entry_Invoices	7.089	10.463	14.269	13.252	3.329
	1	CreateInvoice_select_Suplier_Name	0.118	0.19	0.387	0.139	0.114
	1	iProcurement_Click_AddToCart	12.667	18.735	25.378	23.895	5.926
	1	CreateInvoice_Click_Distributions	1.633	1.972	2.337	1.959	0.249
	1	CreateInvoice_select_taxcode	0.289	0.417	0.619	0.397	0.123
	1	CreateInvoice_select_account	0.784	0.955	1.324	0.892	0.216
	1	CreateInvoice_select_Save	2.183	2.446	3.13	2.29	0.397
	1	CRM_Manage_Role_Click_User_Maintenance	0.191	0.385	0.741	0.404	0.222
	1	CreateInvoice_logout	0.165	0.259	0.49	0.202	0.134
	1	iProcurement_Click_ViewCart	0.274	0.364	0.534	0.367	0.105
	1	CRM_Manage_Role_Search	0.171	0.254	0.37	0.269	0.076
	1	iProcurement_Delete_Item	0.18	0.185	0.191	0.187	0.005
	1	iProcurement_Logout	0.196	0.22	0.26	0.214	0.024
	1	CRM_Manage_Role_Select_Test_user	0.212	0.265	0.314	0.274	0.036
	1	CRM_Manage_Role_Select_Role	0.329	0.359	0.378	0.378	0.021
	1	CRM_Manage_Role_Click_Update	0.299	0.339	0.369	0.369	0.03
	1	CRM_Manage_Role_Logout	0.366	0.392	0.417	0.404	0.02

168352

Expected Results

- 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.

Results

Oracle Apps Traffic from Branch 1 to DCb without WAAS passed.

Oracle Apps 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.

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

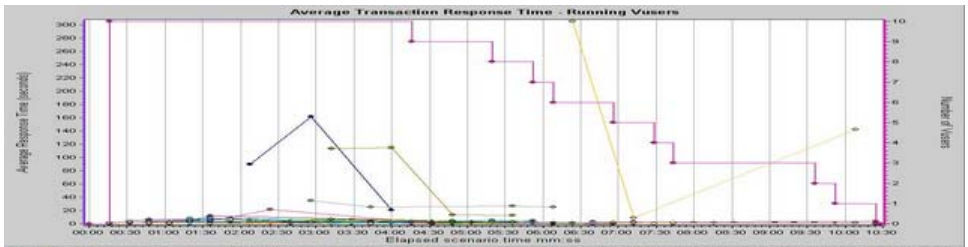
The procedure used to perform the Oracle Apps Traffic from Branch 2 to DCb without WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 3 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 4 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays the average transaction response times relative to the number of vusers running at any given point during the test. The second graph helps determine the percentage of transactions that met predefined response times.



Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev
	Average Transact	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.036	0.113	0.032	0.046
	Average Transact	1	Create_Project_forms_Click_Copy_To	0.644	0.727	0.819	0.719	0.072
	Average Transact	1	Create_Project_forms_Click_find	1.646	2.036	2.291	2.171	0.28
	Average Transact	1	Create_Project_forms_click_open	0.856	1.546	2.784	0.998	0.677
	Average Transact	1	Create_Project_forms_Click_project_link	20.455	90.465	161.156	89.784	57.443
	Average Transact	1	Create_Project_forms_Click_project_name	0.753	1.222	1.688	1.303	0.354
	Average Transact	1	Create_Project_forms_Click_project_vision_service	1.174	4.442	6.515	5.636	2.336
	Average Transact	1	Create_Project_forms_EBusiness_Links	0.443	2.036	3.485	3.398	1.412
	Average Transact	1	Create_Project_forms_Login	1.216	2.517	3.821	2.52	0.921
	Average Transact	1	Create_Project_forms_Logout	0.671	1.386	2.236	1.05	0.606
	Average Transact	1	Create_Project_forms_select_Change_Status	1.087	1.159	1.244	1.143	0.065
	Average Transact	1	Create_Project_forms_select_Customer_name	1.327	1.374	1.414	1.38	0.036
	Average Transact	1	Create_Project_forms_select_organization	1.144	1.278	1.386	1.303	0.101
	Average Transact	1	Create_Project_forms_select_PM	1.304	1.432	1.511	1.48	0.091
	Average Transact	1	CreateInvoice_Click_Apps_Logon_Links	0	0.139	0.334	0.222	0.145
	Average Transact	1	CreateInvoice_Click_Distributions	2.799	3.449	4.549	3.46	0.679
	Average Transact	1	CreateInvoice_Click_EBusiness_Links	0.523	2.473	5.657	2.82	2.035
	Average Transact	1	CreateInvoice_click_Entry_Invoices	12.528	63.511	114.495	113.595	50.536
	Average Transact	1	CreateInvoice_click_Payables_Vision_Operations	2.597	4.467	6.13	6	1.619
	Average Transact	1	CreateInvoice_Login	1.052	3.918	8.452	4.18	2.853
	Average Transact	1	CreateInvoice_logout	0.784	1.642	3.92	1.69	1.245
	Average Transact	1	CreateInvoice_select_account	1.73	2.233	2.431	2.388	0.291
	Average Transact	1	CreateInvoice_select_Save	3.4	4.089	4.627	4.444	0.572
	Average Transact	1	CreateInvoice_select_Suplier_Name	0.312	0.786	1.741	0.563	0.56
	Average Transact	1	CreateInvoice_select_taxcode	1.483	1.551	1.602	1.596	0.051
	Average Transact	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.118	0.353	0	0.166
	Average Transact	1	CRM_Manage_Role_Click_CRM_HTML_Administ	0.676	1.578	3.36	0.699	1.26
	Average Transact	1	CRM_Manage_Role_Click_Setup_Home	0.542	1.794	4.223	0.616	1.718
	Average Transact	1	CRM_Manage_Role_Click_Update	1.076	1.323	1.66	1.425	0.236
	Average Transact	1	CRM_Manage_Role_Click_Use_Maintenance	0.298	2.431	5.846	1.15	2.439
	Average Transact	1	CRM_Manage_Role_Click_Users	0.273	1.168	2.938	0.293	1.252
	Average Transact	1	CRM_Manage_Role_EBusiness_Links	0.101	4.435	12.186	4.708	4.61
	Average Transact	1	CRM_Manage_Role_Login	0.576	2.064	4.359	1.845	1.403
	Average Transact	1	CRM_Manage_Role_Logout	0.426	1.045	1.702	1.046	0.452
	Average Transact	1	CRM_Manage_Role_Search	1.044	1.321	1.619	1.301	0.295
	Average Transact	1	CRM_Manage_Role_Select_Role	1.172	1.321	1.407	1.385	0.106
	Average Transact	1	CRM_Manage_Role_Select_Test_user	1.073	1.226	1.443	1.162	0.169
	Average Transact	1	DCAP_Receivables_Click_LineItems_EnterDet	3.209	3.333	3.486	3.303	0.115
	Average Transact	1	DCAP_Receivables_Click_Transactions	0.908	152.23	305.229	142.554	121.166
	Average Transact	1	DCAP_Receivables_EBusiness_Links	0.09	1.597	3.444	2.185	1.384
	Average Transact	1	DCAP_Receivables_EnterTransDetails_Save	1.183	1.293	1.512	1.185	0.155
	Average Transact	1	DCAP_Receivables_LineItems_Save	0.416	0.496	0.56	0.512	0.06
	Average Transact	1	DCAP_Receivables_Login	0.604	2.627	3.918	3.834	1.364
	Average Transact	1	DCAP_Receivables_LogonLinks	0	1.04	3.806	0.353	1.604
	Average Transact	1	DCAP_Receivables_Logout	0.248	0.265	0.283	0.264	0.014
	Average Transact	1	DCAP_Receivables_ShpTo	1.442	1.513	1.549	1.549	0.051
	Average Transact	1	DCAP_Receivables_Vision_Operation	1.161	3.364	5.764	4.599	1.852
	Average Transact	1	#Procurement_AppsLogonLinks	0	0.13	0.333	0.186	0.139
	Average Transact	1	#Procurement_Choose	3.376	8.471	21.623	4.559	7.606
	Average Transact	1	#Procurement_Click_AddToCart	25.691	27.947	34.46	26.912	3.815
	Average Transact	1	#Procurement_Click_AnkleSuppost	2.685	4.561	6.957	4.912	1.592
	Average Transact	1	#Procurement_Click_Categories	1.368	1.788	2.915	1.492	0.652
	Average Transact	1	#Procurement_Click_Ergonomic_Supplies	0.933	1.553	2.486	1.47	0.573
	Average Transact	1	#Procurement_Click_ViewCart	0.597	0.755	1.075	0.887	0.236
	Average Transact	1	#Procurement_Delete_Item	0.247	0.579	0.913	0.597	0.236
	Average Transact	1	#Procurement_EbusinessHomepage	0.37	4.603	8.697	8.434	3.924
	Average Transact	1	#Procurement_Homepage	0	0.303	0.639	0.575	0.304
	Average Transact	1	#Procurement_Logon	0.988	2.799	5.697	3.058	1.841
	Average Transact	1	#Procurement_Logout	0.282	1.547	3.936	1.429	1.444
	Running Vusers	1	Run	0	4.231	10	4	3.423

168353

Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.036	0.113	0.032	0.046
	1	Create_Project_forms_EBusiness_Links	0.443	2.036	3.485	3.398	1.412
	1	Create_Project_forms_Login	1.216	2.519	3.821	2.52	0.921
	1	Create_Project_forms_Click_project_vision_service	1.174	4.462	6.515	5.636	2.336
	1	DCAP_Receivables_LogonLinks	0.001	1.04	3.806	0.353	1.604
	1	DCAP_Receivables_EBusiness_Links	0.09	1.597	3.444	2.185	1.384
	1	DCAP_Receivables_Login	0.604	2.627	3.918	3.834	1.364
	1	#Procurement_Homepage	0	0.303	0.639	0.575	0.304
	1	CreateInvoice_Click_Apps_Logon_Links	0	0.139	0.334	0.222	0.145
	1	CreateInvoice_Click_EBusiness_Links	0.523	2.473	5.657	2.82	2.035
	1	CreateInvoice_Login	1.052	3.918	8.452	4.18	2.853
	1	#Procurement_AppsLogonLinks	0	0.13	0.333	0.186	0.139
	1	DCAP_Receivables_Vision_Operation	1.161	3.364	5.764	4.599	1.852
	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.177	0.595	0.112	0.246
	1	#Procurement_EbusinessHomepage	0.37	4.603	8.697	8.434	3.924
	1	CreateInvoice_click_Payables_Vision_Operations	2.597	4.467	6.13	6	1.619
	1	#Procurement_Logon	0.988	2.799	5.697	3.058	1.841
	1	CRM_Manage_Role_EBusiness_Links	0.101	4.435	12.186	4.708	4.61
	1	CRM_Manage_Role_Login	0.576	2.064	4.359	1.845	1.403
	1	#Procurement_Choose	3.376	8.471	21.623	4.559	7.606
	1	CRM_Manage_Role_Click_CRM_HTML_Administ	0.676	2.024	5.465	1.255	2
	1	#Procurement_Click_Categories	1.368	1.788	2.915	1.492	0.652
	1	#Procurement_Click_Ergonomic_Supplies	0.933	1.553	2.486	1.47	0.573
	1	Create_Project_forms_Click_project_link	20.455	91.172	161.156	89.784	57.586
	1	#Procurement_Click_AnkleSuppost	2.685	4.561	6.957	4.912	1.592
	1	Create_Project_forms_Click_project_name	0.753	1.225	1.688	1.303	0.354
	1	Create_Project_forms_Click_find	1.646	2.039	2.291	2.171	0.28
	1	Create_Project_forms_Click_Copy_To	0.644	0.728	0.819	0.719	0.072
	1	Create_Project_forms_select_PM	1.304	1.433	1.511	1.48	0.091
	1	Create_Project_forms_select_Customer_name	1.327	1.374	1.414	1.38	0.036
	1	Create_Project_forms_select_organization	1.144	1.279	1.386	1.303	0.101
	1	Create_Project_forms_click_open	0.856	1.559	2.784	0.998	0.681
	1	Create_Project_forms_select_Change_Status	1.087	1.159	1.244	1.143	0.065
	1	Create_Project_forms_Logout	0.671	1.394	2.236	1.05	0.609
	1	CRM_Manage_Role_Click_Setup_Home	0.542	2.401	5.097	3.349	1.924
	1	CRM_Manage_Role_Click_Users	0.273	1.61	3.007	2.869	1.329
	1	#Procurement_Click_AddToCart	25.691	27.947	34.46	26.912	3.815
	1	CRM_Manage_Role_Click_Use_Maintenance	0.298	3.265	9.425	2.266	3.613
	1	#Procurement_Click_ViewCart	0.597	0.755	1.075	0.887	0.236
	1	CreateInvoice_click_Entry_Invoices	12.528	63.511	114.495	113.595	50.536
	1	CreateInvoice_select_Suplier_Name	0.312	0.786	1.741	0.563	0.56
	1	CreateInvoice_Click_Distributions	2.799	3.449	4.549	3.46	0.679
	1	#Procurement_Delete_Item	0.247	0.579	0.913	0.597	0.236
	1	CreateInvoice_select_taxcode	1.483	1.552	1.602	1.602	0.052
	1	CreateInvoice_select_account	1.73	2.233	2.431	2.388	0.291
	1	#Procurement_Logout	0.282	1.547	3.936	1.429	1.444
	1	CRM_Manage_Role_Search	0.983	1.317	1.619	1.619	0.304
	1	CreateInvoice_select_Save	3.4	4.089	4.627	4.444	0.572
	1	CreateInvoice_logout	0.784	1.642	3.92	1.68	1.245
	1	CRM_Manage_Role_Select_Test_user	0.959	1.188	1.443	1.287	0.214
	1	CRM_Manage_Role_Select_Role	1.172	1.343	1.513	1.385	0.124
	1	CRM_Manage_Role_Click_Update	1.076	1.323	1.66	1.425	0.236
	1	CRM_Manage_Role_Logout	0.426	1.045	1.702	1.046	0.452
	1	DCAP_Receivables_Click_Transactions	0.908	153.76	305.229	142.554	121.516
	1	DCAP_Receivables_ShpTo	1.442	1.514	1.549	1.549	0.05
	1	DCAP_Receivables_EnterTransDetails_Save	1.185	1.296	1.512	1.185	0.155
	1	DCAP_Receivables_Click_LineItems_EnterDet	3.209	3.334	3.486	3.303	0.116
	1	DCAP_Receivables_LineItems_Save	0.416	0.497	0.56	0.512	0.06
	1	DCAP_Receivables_Logout	0.248	0.265	0.283	0.264	0.014

168354

Expected Results

- We expect Oracle transactions submitted through Load Runner to be completed successfully
- We expect transaction response times to be slightly higher due to latency and bandwidth contraints and with WAAS being 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

Results

Oracle Apps Traffic from Branch 2 to DCb without WAAS passed.

Oracle Apps 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.

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

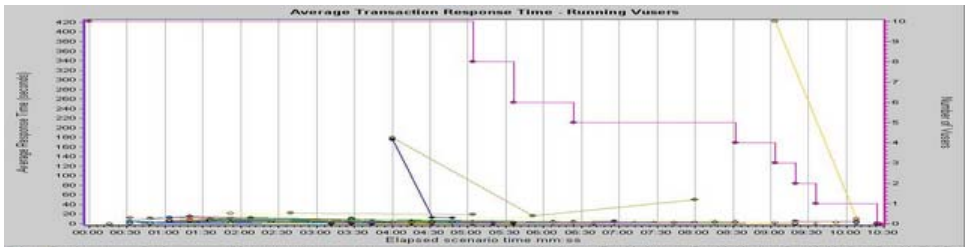
The procedure used to perform the Oracle Apps Traffic from Branch 3 to DCb without WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

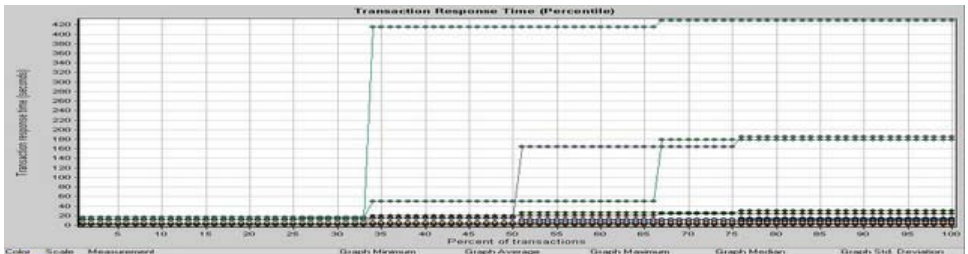
Step 3 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 4 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays 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.



Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev
	Average Transact	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.059	0.155	0.081	0.064
	Average Transact	1	Create_Project_forms_Click_Copy_To	0.703	0.88	1.057	1.057	0.177
	Average Transact	1	Create_Project_forms_Click_find	2.075	2.439	2.804	2.804	0.364
	Average Transact	1	Create_Project_forms_click_open	0.906	1.03	1.113	1.072	0.09
	Average Transact	1	Create_Project_forms_Click_project_link	12.213	66.776	174.714	13.401	76.325
	Average Transact	1	Create_Project_forms_Click_project_name	1.544	1.71	1.877	1.877	0.166
	Average Transact	1	Create_Project_forms_Click_project_vision_service	2.843	9.58	14.906	10.989	5.024
	Average Transact	1	Create_Project_forms_EBusiness_Links	0.449	1.733	4.036	1.187	1.816
	Average Transact	1	Create_Project_forms_Login	1.334	2.062	3.093	1.665	1.065
	Average Transact	1	Create_Project_forms_Logout	1.12	1.303	1.564	1.225	0.189
	Average Transact	1	Create_Project_forms_select_Change_Status	1.443	1.488	1.533	1.533	0.045
	Average Transact	1	Create_Project_forms_select_Customer_name	1.083	1.502	1.857	1.564	0.319
	Average Transact	1	Create_Project_forms_select_organization	1.444	1.567	1.756	1.503	0.136
	Average Transact	1	Create_Project_forms_select_PM	1.498	1.593	1.726	1.653	0.097
	Average Transact	1	CreateInvoice_Click_Apps_Logon_Links	0	0.163	0.5	0.151	0.204
	Average Transact	1	CreateInvoice_Click_Distributions	3.726	4.644	5.776	4.427	0.952
	Average Transact	1	CreateInvoice_Click_EBusiness_Links	1.537	3.092	5.436	3.822	1.64
	Average Transact	1	CreateInvoice_click_Entry_Invoices	16.768	82.031	179.189	50.137	70.038
	Average Transact	1	CreateInvoice_click_Payables_Vision_Operations	2.374	5.207	8.03	7.155	2.848
	Average Transact	1	CreateInvoice_Login	1.989	3.679	5.411	5.175	1.47
	Average Transact	1	CreateInvoice_logout	0.996	1.182	1.292	1.259	0.132
	Average Transact	1	CreateInvoice_select_account	2.983	3.129	3.248	3.156	0.11
	Average Transact	1	CreateInvoice_select_Save	4.302	4.427	4.819	4.312	0.242
	Average Transact	1	CreateInvoice_select_Suplier_Name	0.549	0.767	1.013	0.739	0.19
	Average Transact	1	CreateInvoice_select_taxcode	1.773	1.967	2.126	2.001	0.146
	Average Transact	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.083	0.175	0.157	0.083
	Average Transact	1	CRM_Manage_Role_Click_CRM_HTML_Administ	2.495	4.655	8.247	3.224	2.557
	Average Transact	1	CRM_Manage_Role_Click_Setup_Home	1.642	5.843	12.384	6.83	4.256
	Average Transact	1	CRM_Manage_Role_Click_Update	1.421	1.863	2.547	1.621	0.49
	Average Transact	1	CRM_Manage_Role_Click_User_Maintenance	1.562	2.622	3.376	2.927	0.771
	Average Transact	1	CRM_Manage_Role_Click_Users	1.05	2.5	3.951	3.416	1.213
	Average Transact	1	CRM_Manage_Role_EBusiness_Links	0.735	1.38	1.765	1.639	0.459
	Average Transact	1	CRM_Manage_Role_Login	1.409	2.829	5.697	2.453	1.698
	Average Transact	1	CRM_Manage_Role_Logout	0.884	1.456	2.316	1.427	0.532
	Average Transact	1	CRM_Manage_Role_Search	2.547	3.313	4.719	3	0.632
	Average Transact	1	CRM_Manage_Role_Select_Role	1.306	4.555	11.393	3.121	4
	Average Transact	1	CRM_Manage_Role_Select_Test_user	1.803	3.996	6.996	2.924	2.914
	Average Transact	1	DCAP_Receivables_Click_LineItems_EnterDet	5.358	5.414	5.47	5.47	0.056
	Average Transact	1	DCAP_Receivables_Click_Transactions	10.671	286.304	421.993	421.993	205.661
	Average Transact	1	DCAP_Receivables_EBusiness_Links	0.241	1.201	1.938	1.424	0.71
	Average Transact	1	DCAP_Receivables_EnterTransDetails_Save	1.244	1.299	1.353	1.353	0.055
	Average Transact	1	DCAP_Receivables_LineItems_Save	0.545	0.547	0.548	0.548	0.001
	Average Transact	1	DCAP_Receivables_Login	0.948	6.56	12.027	6.706	4.524
	Average Transact	1	DCAP_Receivables_LogonLinks	0	0.309	0.473	0.454	0.210
	Average Transact	1	DCAP_Receivables_Logout	0.636	0.673	0.71	0.71	0.037
	Average Transact	1	DCAP_Receivables_ShipTo	2.04	2.214	2.387	2.387	0.174
	Average Transact	1	DCAP_Receivables_Vision_Operation	1.552	11.594	21.637	21.637	10.043
	Average Transact	1	#Procurement_AppsLogonLinks	0	0.102	0.225	0.079	0.093
	Average Transact	1	#Procurement_Choose	2.575	9.474	13.838	12.01	4.935
	Average Transact	1	#Procurement_Click_AddToCart	19.516	21.113	22.709	22.709	1.596
	Average Transact	1	#Procurement_Click_AnkleSuppot	2.797	6.634	10.937	10.486	3.883
	Average Transact	1	#Procurement_Click_Categories	2.083	3.889	7.076	2.268	2.311
	Average Transact	1	#Procurement_Click_Ergonomic_Supplies	1.195	2.923	6.666	2.132	2.186
	Average Transact	1	#Procurement_Click_ViewCart	0.646	1.243	1.839	1.639	0.596
	Average Transact	1	#Procurement_Delete_Item	0.767	0.88	1.055	0.818	0.126
	Average Transact	1	#Procurement_EbusinessHomepage	1.096	1.693	2.038	1.946	0.424
	Average Transact	1	#Procurement_Homepage	0	0.143	0.428	0	0.202
	Average Transact	1	#Procurement_Logon	2.172	6.603	12.892	4.745	4.57
	Average Transact	1	#Procurement_Logout	1.19	1.555	2.185	1.291	0.447
	Running Vusers	1	Run	0	3.9	10	4	3.268

168355



Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	#Procurement_Homepage	0	0.214	0.465	0.392	0.216
	1	DCAP_Receivables_LogonLinks	0	0.311	0.473	0.454	0.218
	1	DCAP_Receivables_EBusiness_Links	0.241	1.208	1.938	1.424	0.711
	1	CreateInvoice_Click_Apps_Logon_Links	0	0.163	0.5	0.151	0.204
	1	#Procurement_AppsLogonLinks	0	0.076	0.225	0.079	0.092
	1	CreateInvoice_Click_EBusiness_Links	1.537	3.092	5.436	3.822	1.64
	1	#Procurement_EbusinessHomepage	0.933	1.544	2.038	1.946	0.464
	1	DCAP_Receivables_Login	0.948	6.615	12.027	6.706	4.534
	1	CreateInvoice_Login	1.989	3.679	5.411	5.175	1.47
	1	#Procurement_Logon	2.169	5.495	12.892	4.745	4.398
	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.059	0.155	0.081	0.064
	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.083	0.175	0.157	0.083
	1	Create_Project_forms_EBusiness_Links	0.449	1.733	4.036	1.187	1.816
	1	CRM_Manage_Role_EBusiness_Links	0.735	1.445	2.012	1.765	0.49
	1	Create_Project_forms_Login	1.334	2.062	3.093	1.665	1.065
	1	CRM_Manage_Role_Login	1.409	2.829	5.697	2.453	1.698
	1	Create_Project_forms_Click_project_vision_service	2.129	7.895	14.906	10.989	5.263
	1	CreateInvoice_click_Payables_Vision_Operations	2.374	5.207	8.03	7.155	2.848
	1	#Procurement_Choose	2.392	7.749	13.838	12.01	5.216
	1	CRM_Manage_Role_Click_CRM_HTML_Administ	2.495	5.553	9.942	6.551	2.96
	1	#Procurement_Click_Categories	1.309	3.424	7.076	3.228	2.216
	1	#Procurement_Click_Ergonomic_Supplies	1.195	2.923	6.666	2.132	2.186
	1	CRM_Manage_Role_Click_Setup_Home	1.642	5.843	12.384	6.83	4.256
	1	DCAP_Receivables_Vision_Operation	1.552	15.626	23.417	19.057	3.568
	1	CRM_Manage_Role_Click_Users	1.05	2.5	3.951	3.416	1.213
	1	#Procurement_Click_AnkleSuppot	2.797	6.834	10.937	10.486	3.883
	1	CRM_Manage_Role_Click_User_Maintenance	1.023	2.357	3.376	2.927	0.895
	1	#Procurement_Click_AddToCart	12.552	21.113	30.25	26.491	7.369
	1	#Procurement_Click_ViewCart	0.529	1.243	2.644	1.034	0.629
	1	CRM_Manage_Role_Search	2.547	3.313	4.719	3	0.632
	1	CRM_Manage_Role_Select_Test_user	1.803	3.996	6.996	2.924	2.914
	1	#Procurement_Delete_Item	0.733	0.924	1.379	0.818	0.264
	1	#Procurement_Logout	1.19	1.713	2.292	2.079	0.48
	1	CRM_Manage_Role_Select_Role	1.306	4.555	11.393	3.121	4
	1	Create_Project_forms_Click_project_link	12.213	93.761	184.735	164.694	81.264
	1	Create_Project_forms_Click_project_name	1.303	1.71	1.896	1.857	0.239
	1	CreateInvoice_click_Entry_Invoices	16.768	83.003	179.189	50.137	70.354
	1	Create_Project_forms_Click_find	1.674	2.439	2.934	2.714	0.396
	1	Create_Project_forms_Click_Copy_To	0.606	0.88	1.08	1.034	0.191
	1	Create_Project_forms_select_PM	1.498	1.626	1.793	1.66	0.113
	1	Create_Project_forms_select_Customer_name	1.083	1.591	1.86	1.853	0.316
	1	CreateInvoice_select_Suplier_Name	0.549	0.77	1.013	0.739	0.191
	1	Create_Project_forms_select_organization	1.444	1.615	1.773	1.739	0.143
	1	CRM_Manage_Role_Click_Update	1.421	1.882	2.547	1.71	0.442
	1	Create_Project_forms_click_open	0.906	1.051	1.141	1.085	0.098
	1	Create_Project_forms_select_Change_Status	1.488	1.488	1.569	1.497	0.057
	1	CreateInvoice_Click_Distributions	3.726	4.655	5.776	4.427	0.855
	1	Create_Project_forms_Logout	1.076	1.258	1.564	1.225	0.185
	1	CreateInvoice_select_taxcode	1.773	1.968	2.126	2.001	0.146
	1	CreateInvoice_select_account	2.983	3.13	3.248	3.156	0.11
	1	CreateInvoice_select_Save	4.302	4.481	4.819	4.312	0.243
	1	CreateInvoice_logout	0.996	1.183	1.292	1.259	0.132
	1	CRM_Manage_Role_Logout	0.884	1.456	2.316	1.427	0.532
	1	DCAP_Receivables_Click_Transactions	10.671	286.304	428.683	415.304	193.534
	1	DCAP_Receivables_ShipTo	2.04	2.214	2.387	2.387	0.174
	1	DCAP_Receivables_EnterTransDetails_Save	1.244	1.299	1.353	1.353	0.055
	1	DCAP_Receivables_Click_LineItems_EnterDet	5.358	5.414	5.47	5.47	0.056
	1	DCAP_Receivables_LineItems_Save	0.548	0.548	0.548	0.548	0
	1	DCAP_Receivables_Logout	0.636	0.673	0.71	0.71	0.037

168356

Expected Results

- We expect Oracle transactions submitted through Load Runner to 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 time out due to high latency and bandwidth.
- 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.

Results

Oracle Apps Traffic from Branch 3 to DCb without WAAS passed.

E-Biz Branches to DCb with WAAS

E-Biz Branches to DCb with WAAS verifies configuration and functionality of Oracle E-Business Application deployment in Data Center A. These tests validate integration of the Application with GSS and CSM. Application traffic is simulated using Mercury Load runner from all the 3 branch clients. GSS is configured to be authoritative for the wwwin-oefin domain which resolved client requests to the VIP physically located in DCb

WAAS is Enabled to observe the Application traffic optimization.

The following tests were performed:

- Oracle Apps Traffic from Branch 1 to DCb with WAAS, page 8-41
- Oracle Apps Traffic from Branch 2 to DCb with WAAS, page 8-43
- Oracle Apps Traffic from Branch 3 to DCb with WAAS, page 8-46

Oracle Apps 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.

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 Apps Traffic from Branch 1 to DCb with WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

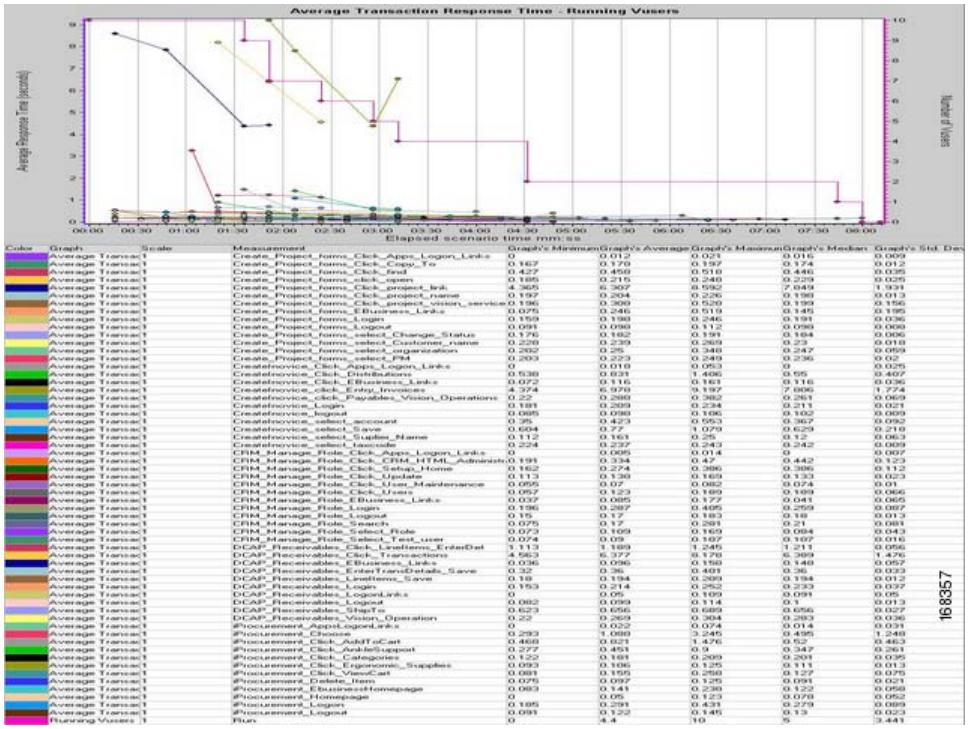
Step 3 Clear the appropriate counters on the WAAS devices by issuing the **clear statistics all** command.

Step 4 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 5 Verify the traffic is being accelerated by the WAE devices at the branch by issuing the **show tfo connection summary** command.

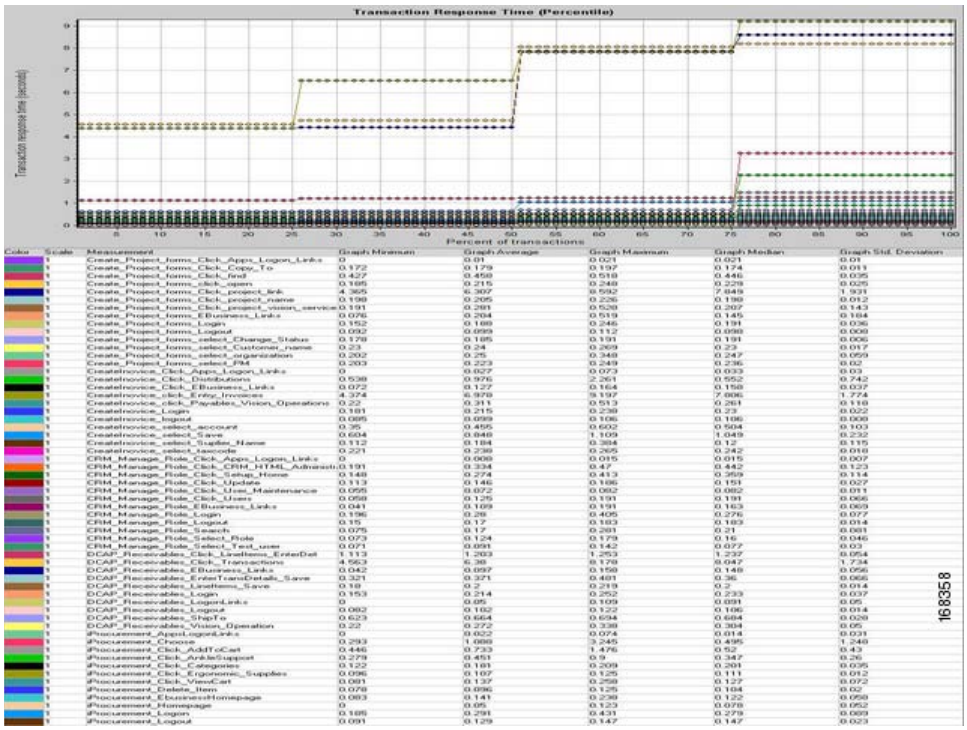
From the output of the command you should see all client to application connections established and being fully optimized.

Step 6 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays 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.



Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev
	Average Transact		Create_Project_forms_Click_Apps_Logon_Links	0	0.012	0.021	0.016	0.009
	Average Transact		Create_Project_forms_Click_Copy_To	0.167	0.179	0.197	0.174	0.012
	Average Transact		Create_Project_forms_Click_find	0.427	0.459	0.510	0.446	0.035
	Average Transact		Create_Project_forms_click_open	0.185	0.215	0.240	0.229	0.025
	Average Transact		Create_Project_forms_Click_project_link	4.365	6.307	8.592	7.049	1.931
	Average Transact		Create_Project_forms_Click_project_name	0.197	0.204	0.226	0.198	0.013
	Average Transact		Create_Project_forms_Click_project_vision_service	0.196	0.369	0.520	0.199	0.156
	Average Transact		Create_Project_forms_EBusiness_Links	0.075	0.246	0.519	0.145	0.195
	Average Transact		Create_Project_forms_Login	0.159	0.198	0.246	0.191	0.036
	Average Transact		Create_Project_forms_Logout	0.091	0.098	0.112	0.098	0.008
	Average Transact		Create_Project_forms_select_Change_Status	0.176	0.182	0.191	0.184	0.006
	Average Transact		Create_Project_forms_select_Customer_name	0.228	0.239	0.269	0.23	0.018
	Average Transact		Create_Project_forms_select_organization	0.282	0.25	0.348	0.247	0.059
	Average Transact		Create_Project_forms_select_PM	0.203	0.223	0.249	0.236	0.02
	Average Transact		CreateInvoice_Click_Apps_Logon_Links	0	0.018	0.053	0	0.025
	Average Transact		CreateInvoice_Click_Distributions	0.538	0.831	1.406	0.55	0.407
	Average Transact		CreateInvoice_Click_EBusiness_Links	0.072	0.116	0.161	0.116	0.036
	Average Transact		CreateInvoice_click_Entry_Invoices	4.374	6.978	9.197	7.806	1.774
	Average Transact		CreateInvoice_click_Payables_Vision_Operations	0.22	0.288	0.382	0.261	0.069
	Average Transact		CreateInvoice_Login	0.181	0.209	0.234	0.211	0.021
	Average Transact		CreateInvoice_logout	0.088	0.098	0.106	0.102	0.009
	Average Transact		CreateInvoice_select_account	0.35	0.423	0.553	0.367	0.092
	Average Transact		CreateInvoice_select_Save	0.684	0.77	1.079	0.629	0.218
	Average Transact		CreateInvoice_select_Suplier_Name	0.112	0.161	0.25	0.12	0.063
	Average Transact		CreateInvoice_select_taxcode	0.224	0.237	0.243	0.242	0.009
	Average Transact		CRM_Manage_Role_Click_Apps_Logon_Links	0	0.006	0.014	0	0.007
	Average Transact		CRM_Manage_Role_Click_CRM_HTML_Administ	0.191	0.334	0.47	0.442	0.123
	Average Transact		CRM_Manage_Role_Click_Setup_Home	0.162	0.274	0.386	0.386	0.112
	Average Transact		CRM_Manage_Role_Click_Update	0.113	0.138	0.169	0.133	0.023
	Average Transact		CRM_Manage_Role_Click_User_Maintenance	0.055	0.07	0.082	0.074	0.01
	Average Transact		CRM_Manage_Role_Click_Users	0.057	0.123	0.189	0.189	0.066
	Average Transact		CRM_Manage_Role_EBusiness_Links	0.037	0.085	0.177	0.041	0.065
	Average Transact		CRM_Manage_Role_Login	0.196	0.287	0.495	0.259	0.087
	Average Transact		CRM_Manage_Role_Logout	0.15	0.17	0.183	0.18	0.013
	Average Transact		CRM_Manage_Role_Search	0.075	0.17	0.281	0.21	0.081
	Average Transact		CRM_Manage_Role_Select_Role	0.073	0.109	0.169	0.084	0.043
	Average Transact		CRM_Manage_Role_Select_Test_user	0.074	0.09	0.107	0.107	0.016
	Average Transact		DCAP_Receivables_Click_LineItems_EnterDet	1.113	1.189	1.245	1.211	0.056
	Average Transact		DCAP_Receivables_Click_Transactions	4.583	5.377	6.178	5.389	1.476
	Average Transact		DCAP_Receivables_EBusiness_Links	0.036	0.096	0.159	0.148	0.057
	Average Transact		DCAP_Receivables_EnterTransDetails_Save	0.32	0.36	0.401	0.36	0.033
	Average Transact		DCAP_Receivables_LineItems_Save	0.18	0.194	0.269	0.194	0.012
	Average Transact		DCAP_Receivables_Login	0.153	0.214	0.252	0.233	0.037
	Average Transact		DCAP_Receivables_LogonLinks	0	0.05	0.109	0.091	0.05
	Average Transact		DCAP_Receivables_Logout	0.082	0.099	0.114	0.1	0.013
	Average Transact		DCAP_Receivables_ShipTo	0.623	0.656	0.689	0.656	0.027
	Average Transact		DCAP_Receivables_Vision_Operation	0.22	0.269	0.304	0.283	0.036
	Average Transact		Procurement_AppsLogonLinks	0	0.022	0.074	0.014	0.031
	Average Transact		Procurement_Choose	0.293	1.088	3.245	0.495	1.248
	Average Transact		Procurement_Click_AddToCart	0.468	0.821	1.476	0.52	0.463
	Average Transact		Procurement_Click_AnkleSupport	0.277	0.453	0.9	0.347	0.261
	Average Transact		Procurement_Click_Categories	0.122	0.181	0.269	0.201	0.035
	Average Transact		Procurement_Click_Ergonomic_Supplies	0.093	0.106	0.125	0.111	0.013
	Average Transact		Procurement_Click_ViewCart	0.081	0.155	0.258	0.127	0.075
	Average Transact		Procurement_Delete_Item	0.075	0.097	0.125	0.091	0.021
	Average Transact		Procurement_EBusinessHomepage	0.083	0.141	0.238	0.122	0.058
	Average Transact		Procurement_Homepage	0	0.05	0.123	0.078	0.052
	Average Transact		Procurement_Login	0.185	0.291	0.431	0.279	0.089
	Average Transact		Procurement_Logout	0.091	0.122	0.145	0.13	0.023
	Running Vusers		Run	0	4.4	10	5	3.441

168357



Step 7 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
```

Expected Results

- We expect WAAS to accelerate transaction response times.
- We expect transaction response times to be slightly higher in comparison to DCa.
- 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.

Results

Oracle Apps Traffic from Branch 1 to DCb with WAAS passed.

Oracle Apps 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.

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 Apps Traffic from Branch 2 to DCb with WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

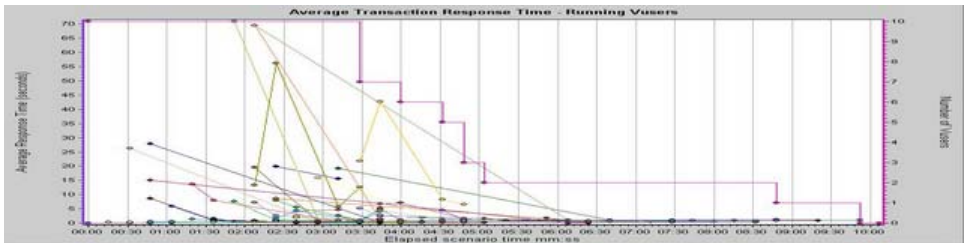
Step 3 Clear the appropriate counters on the WAAS devices by issuing the **clear statistics all** command.

Step 4 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 5 Verify the traffic is being accelerated by the WAE devices at the branch by issuing the **show tfo connection summary** command.

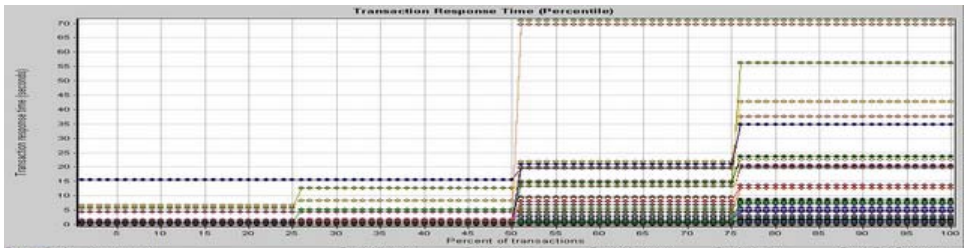
From the output of the command you should see all client to application connections established and being fully optimized.

Step 6 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays 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.



Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev
	Average Transac	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.039	0.064	0.054	0.020
	Average Transac	1	Create_Project_forms_Click_Copy_To	0.349	0.456	0.563	0.563	0.107
	Average Transac	1	Create_Project_forms_Click_find	4.263	4.476	4.689	4.689	0.213
	Average Transac	1	Create_Project_forms_click_open	0.289	0.52	0.75	0.75	0.231
	Average Transac	1	Create_Project_forms_Click_project_link	15.537	17.704	19.87	19.87	2.166
	Average Transac	1	Create_Project_forms_Click_project_name	0.414	0.556	0.697	0.697	0.142
	Average Transac	1	Create_Project_forms_Click_project_vision_services	6.36	10.134	19.633	19.633	9.499
	Average Transac	1	Create_Project_forms_EBusiness_Links	0.113	13.182	26.25	26.25	13.069
	Average Transac	1	Create_Project_forms_Login	1.124	36.032	70.94	70.94	34.908
	Average Transac	1	Create_Project_forms_Logout	0.726	0.883	1.039	1.039	0.156
	Average Transac	1	Create_Project_forms_select_Change_Status	0.522	0.636	0.749	0.749	0.114
	Average Transac	1	Create_Project_forms_select_Customer_name	0.874	1.682	2.49	2.49	0.808
	Average Transac	1	Create_Project_forms_select_organization	0.431	0.472	0.513	0.513	0.041
	Average Transac	1	Create_Project_forms_select_PM	0.433	0.722	1.011	1.011	0.289
	Average Transac	1	CreateInvoice_Click_Apps_Logon_Links	0	0.103	0.25	0.059	0.106
	Average Transac	1	CreateInvoice_Click_Distributions	0.932	4.91	7.953	5.584	2.53
	Average Transac	1	CreateInvoice_Click_EBusiness_Links	0.117	3.289	6.637	1.113	3.893
	Average Transac	1	CreateInvoice_click_Entry_Invoices	5.664	21.99	56.275	13.353	20.022
	Average Transac	1	CreateInvoice_click_Payables_Vision_Operations	0.591	0.989	1.392	0.683	0.356
	Average Transac	1	CreateInvoice_Login	0.316	2.369	5.94	0.851	2.535
	Average Transac	1	CreateInvoice_logout	0.181	0.259	0.408	0.184	0.106
	Average Transac	1	CreateInvoice_select_account	0.592	1.292	1.931	1.611	0.539
	Average Transac	1	CreateInvoice_select_Save	0.771	2.756	4.35	4.099	1.516
	Average Transac	1	CreateInvoice_select_Supplier_Name	0.343	0.562	0.842	0.712	0.22
	Average Transac	1	CreateInvoice_select_taxcode	0.354	0.501	0.633	0.539	0.101
	Average Transac	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.031	0.075	0.048	0.032
	Average Transac	1	CRM_Manage_Role_Click_CRM_HTML_Administ	0.521	3.241	8.573	0.63	3.77
	Average Transac	1	CRM_Manage_Role_Click_Setup_Home	0.969	7.034	19.159	0.974	8.574
	Average Transac	1	CRM_Manage_Role_Click_Update	0.61	1.027	1.513	0.926	0.284
	Average Transac	1	CRM_Manage_Role_Click_User_Maintenance	0.52	1.162	2.484	0.532	0.928
	Average Transac	1	CRM_Manage_Role_Click_Users	0.477	1.16	2.664	0.979	0.89
	Average Transac	1	CRM_Manage_Role_EBusiness_Links	0.099	5.119	15.067	0.191	7.034
	Average Transac	1	CRM_Manage_Role_Login	0.953	23.839	69.625	1.04	32.376
	Average Transac	1	CRM_Manage_Role_Logout	1.007	1.207	1.669	1.132	0.271
	Average Transac	1	CRM_Manage_Role_Search	0.397	0.527	0.61	0.487	0.166
	Average Transac	1	CRM_Manage_Role_Select_Role	0.852	1.876	4.529	1.188	1.537
	Average Transac	1	CRM_Manage_Role_Select_Test_user	0.682	1.17	1.032	1.455	0.492
	Average Transac	1	DCAP_Receivables_Click_LineItems_EnterDet	1.637	5.162	7.169	6.71	2.521
	Average Transac	1	DCAP_Receivables_Click_Transactions	6.658	19.867	42.786	21.76	14.474
	Average Transac	1	DCAP_Receivables_EBusiness_Links	0.105	9.363	27.871	0.112	13.088
	Average Transac	1	DCAP_Receivables_EnterTransDetails_Save	0.489	2.37	4.251	4.251	1.881
	Average Transac	1	DCAP_Receivables_LineItems_Save	0.224	2.076	5.016	0.985	2.104
	Average Transac	1	DCAP_Receivables_Login	0.849	23.651	69.558	1.147	32.32
	Average Transac	1	DCAP_Receivables_LogonLinks	0	0.153	0.34	0.272	0.155
	Average Transac	1	DCAP_Receivables_Logout	0.489	0.707	1.132	0.499	0.301
	Average Transac	1	DCAP_Receivables_ShipTo	0.607	2.921	5.123	2.833	1.763
	Average Transac	1	DCAP_Receivables_Vision_Operation	0.814	9.35	15.887	15.887	7.537
	Average Transac	1	iProcurement_AppsLogonLinks	0	0.024	0.072	0.025	0.029
	Average Transac	1	iProcurement_Choose	0.642	5.828	13.589	7.926	5.324
	Average Transac	1	iProcurement_Click_AddToCart	0.703	2.829	7.28	2.617	2.685
	Average Transac	1	iProcurement_Click_AnkleSupport	0.38	2.385	7.606	1.125	3.029
	Average Transac	1	iProcurement_Click_Categories	0.191	0.619	1.529	0.507	0.539
	Average Transac	1	iProcurement_Click_Ergonomic_Supplies	0.147	0.267	0.446	0.295	0.117
	Average Transac	1	iProcurement_Click_ViewCart	0.123	0.422	0.99	0.419	0.348
	Average Transac	1	iProcurement_Delete_Item	0.094	0.139	0.204	0.151	0.043
	Average Transac	1	iProcurement_EbusinessHomepage	0.063	0.251	0.462	0.244	0.142
	Average Transac	1	iProcurement_Homepage	0	0.116	0.343	0.121	0.14
	Average Transac	1	iProcurement_Logon	0.294	0.422	0.51	0.484	0.084
	Average Transac	1	iProcurement_Logout	0.16	0.204	0.259	0.233	0.043
	RunningVusers	1	Run	0	3.779	10	3	3.258

168359



Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	Create_Project_forms_Click_Apps_Logon_Links	0	0.064	0.064	0.054	0.03
	1	Create_Project_forms_Click_Copy_To	0.352	0.457	0.65	0.476	0.122
	1	Create_Project_forms_Click_find	4.258	4.476	4.03	4.547	0.295
	1	Create_Project_forms_click_open	0.284	0.52	0.774	0.726	0.231
	1	Create_Project_forms_Click_project_link	15.511	17.704	19.966	19.774	2.168
	1	Create_Project_forms_Click_project_name	0.404	0.556	0.703	0.692	0.142
	1	Create_Project_forms_Click_project_vision_service	0.631	10.134	19.224	19.543	9.499
	1	Create_Project_forms_EBusiness_Links	0.112	13.182	37.512	14.989	15.304
	1	Create_Project_forms_Login	1.111	36.032	70.955	70.925	34.908
	1	Create_Project_forms_Logout	0.681	0.883	1.063	1.015	0.161
	1	Create_Project_forms_select_Change_Status	0.517	0.636	0.765	0.733	0.114
	1	Create_Project_forms_select_Customer_name	0.769	1.683	2.491	2.491	0.812
	1	Create_Project_forms_select_organization	0.424	0.472	0.513	0.513	0.041
	1	Create_Project_forms_select_PM	0.431	0.722	1.011	1.011	0.289
	1	CreateInvoice_Click_Apps_Logon_Links	0	0.077	0.25	0.059	0.102
	1	CreateInvoice_Click_Distributions	0.932	4.91	7.953	5.584	2.53
	1	CreateInvoice_Click_EBusiness_Links	0.066	2.496	6.637	1.113	3.569
	1	CreateInvoice_click_Entry_Invoices	5.664	21.99	56.275	13.353	20.022
	1	CreateInvoice_click_Payables_Vision_Operations	0.591	0.937	1.392	0.76	0.327
	1	CreateInvoice_Login	0.316	1.856	5.94	0.851	2.368
	1	CreateInvoice_logout	0.181	0.295	0.552	0.263	0.152
	1	CreateInvoice_select_account	0.592	1.292	1.931	1.611	0.539
	1	CreateInvoice_select_Save	0.771	2.756	4.35	4.099	1.516
	1	CreateInvoice_select_Supplier_Name	0.343	0.562	0.842	0.712	0.22
	1	CreateInvoice_select_taxcode	0.354	0.501	0.633	0.539	0.101
	1	CRM_Manage_Role_Click_Apps_Logon_Links	0	0.031	0.075	0.048	0.032
	1	CRM_Manage_Role_Click_CRM_HTML_Administ	0.521	4.574	12.584	4.581	4.894
	1	CRM_Manage_Role_Click_Setup_Home	0.974	10.066	23.713	14.604	9.646
	1	CRM_Manage_Role_Click_Update	0.61	1.027	1.513	0.926	0.284
	1	CRM_Manage_Role_Click_User_Maintenance	0.52	1.51	3.192	1.796	1.101
	1	CRM_Manage_Role_Click_Users	0.477	1.16	2.664	0.979	0.89
	1	CRM_Manage_Role_EBusiness_Links	0.099	7.606	20.662	9.471	8.445
	1	CRM_Manage_Role_Login	0.953	35.286	69.625	69.625	34.339
	1	CRM_Manage_Role_Logout	1.007	1.207	1.669	1.132	0.271
	1	CRM_Manage_Role_Search	0.397	0.527	0.61	0.487	0.166
	1	CRM_Manage_Role_Select_Role	0.852	1.876	4.529	1.188	1.537
	1	CRM_Manage_Role_Select_Test_user	0.682	1.17	1.032	1.455	0.492
	1	DCAP_Receivables_Click_LineItems_EnterDet	1.539	4.273	6.71	6.71	2.672
	1	DCAP_Receivables_Click_Transactions	6.658	19.867	42.786	21.76	14.474
	1	DCAP_Receivables_EBusiness_Links	0.112	13.992	34.749	20.995	14.707
	1	DCAP_Receivables_EnterTransDetails_Save	0.387	2.37	5.245	3.257	2.009
	1	DCAP_Receivables_LineItems_Save	0.194	1.613	5.016	0.985	1.99
	1	DCAP_Receivables_Login	0.849	35.278	69.566	69.55	34.28
	1	DCAP_Receivables_LogonLinks	0	0.153	0.34	0.272	0.155
	1	DCAP_Receivables_Logout	0.407	0.653	1.132	0.572	0.283
	1	DCAP_Receivables_ShipTo	0.606	2.392	5.123	2.833	1.76
	1	DCAP_Receivables_Vision_Operation	0.76	8.35	22.754	9.02	8.966
	1	iProcurement_AppsLogonLinks	0	0.024	0.072	0.025	0.029
	1	iProcurement_Choose	0.642	5.828	13.589	7.926	5.324
	1	iProcurement_Click_AddToCart	0.703	2.829	7.28	2.617	2.685
	1	iProcurement_Click_AnkleSupport	0.38	2.385	7.606	1.125	3.029
	1	iProcurement_Click_Categories	0.191	0.619	1.529	0.507	0.539
	1	iProcurement_Click_Ergonomic_Supplies	0.147	0.267	0.446	0.295	0.117
	1	iProcurement_Click_ViewCart	0.123	0.422	0.99	0.419	0.348
	1	iProcurement_Delete_Item	0.094	0.139	0.204	0.151	0.043
	1	iProcurement_EbusinessHomepage	0.063	0.251	0.462	0.244	0.142
	1	iProcurement_Homepage	0	0.116	0.343	0.121	0.14
	1	iProcurement_Logon	0.294	0.422	0.51	0.484	0.084
	1	iProcurement_Logout	0.16	0.204	0.259	0.233	0.043

168360

Step 7 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
```

Expected Results

- We expect WAAS to accelerate transaction response times.
- We expect transaction response times to be slightly higher in comparison to DCa.
- 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.

Results

Oracle Apps Traffic from Branch 2 to DCb with WAAS passed.

Oracle Apps 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.

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 Apps Traffic from Branch 3 to DCb with WAAS test follows:

Step 1 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.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

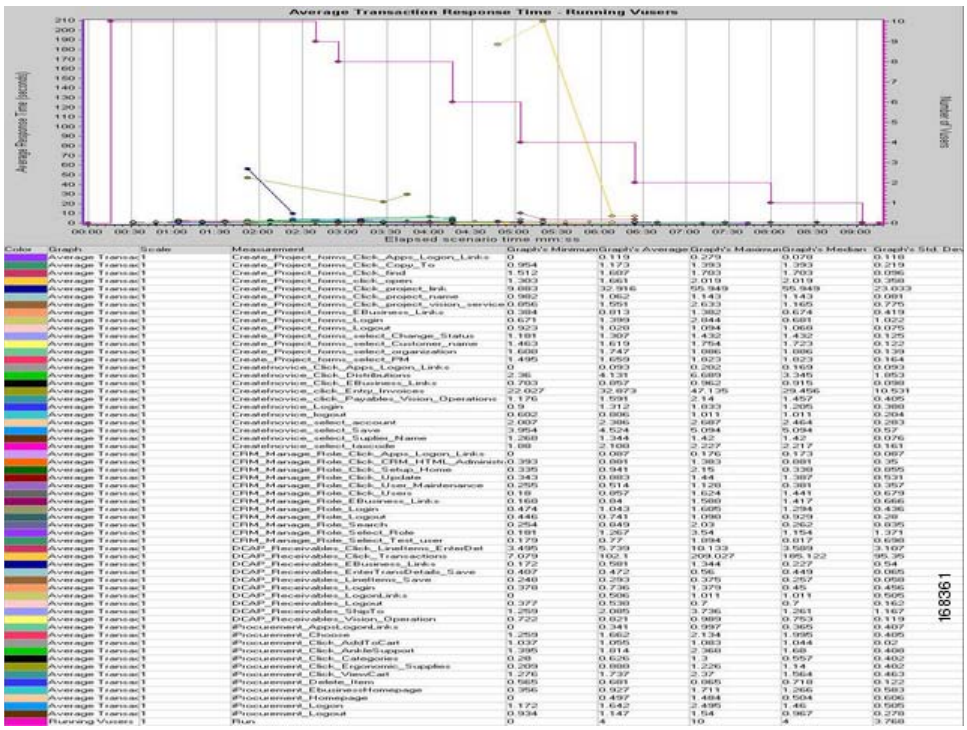
Step 3 Clear the appropriate counters on the WAAS devices by issuing the **clear statistics all** command.

Step 4 Initiate the Load Runner generated traffic which will run for approximately 10 minutes.

Step 5 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 6 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays 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.



Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev
	Average Transac	1	Create_Project_forms_Clck_Apps_Logon_Links	0	0.119	0.279	0.070	0.116
	Average Transac	1	Create_Project_forms_Clck_Copy_To	0.954	1.173	1.393	1.393	0.219
	Average Transac	1	Create_Project_forms_Clck_find	1.512	1.607	1.703	1.703	0.096
	Average Transac	1	Create_Project_forms_clck_open	1.393	1.661	2.019	2.019	0.358
	Average Transac	1	Create_Project_forms_Clck_project_lnk	9.683	32.916	56.949	56.949	23.033
	Average Transac	1	Create_Project_forms_Clck_project_name	0.982	1.062	1.143	1.143	0.081
	Average Transac	1	Create_Project_forms_Clck_project_vision_service	0.856	1.551	2.633	1.165	0.776
	Average Transac	1	Create_Project_forms_EBusiness_Links	0.384	0.813	1.382	0.674	0.419
	Average Transac	1	Create_Project_forms_Login	0.671	1.399	2.844	0.681	1.022
	Average Transac	1	Create_Project_forms_Logout	0.923	1.020	1.094	1.068	0.075
	Average Transac	1	Create_Project_forms_select_Change_Status	1.181	1.307	1.432	1.432	0.125
	Average Transac	1	Create_Project_forms_select_Customer_name	1.493	1.615	1.754	1.723	0.122
	Average Transac	1	Create_Project_forms_select_organization	1.638	1.747	1.886	1.886	0.139
	Average Transac	1	Create_Project_forms_select_PM	1.495	1.659	1.823	1.823	0.164
	Average Transac	1	Createinvoice_Clck_Apps_Logon_Links	0	0.093	0.202	0.169	0.093
	Average Transac	1	Createinvoice_Clck_Distributions	2.36	4.131	6.689	3.345	1.053
	Average Transac	1	Createinvoice_Clck_EBusiness_Links	0.703	0.857	0.962	0.915	0.098
	Average Transac	1	Createinvoice_clck_Entp_Invoices	22.027	32.673	47.135	29.456	10.531
	Average Transac	1	Createinvoice_clck_Payables_Vision_Operations	1.176	1.591	2.14	1.457	0.405
	Average Transac	1	Createinvoice_Login	0.9	1.312	1.633	1.265	0.388
	Average Transac	1	Createinvoice_logout	0.692	0.896	1.011	1.011	0.204
	Average Transac	1	Createinvoice_select_account	2.007	2.386	2.687	2.464	0.283
	Average Transac	1	Createinvoice_select_Save	3.954	4.524	5.094	5.094	0.57
	Average Transac	1	Createinvoice_select_Supplier_Name	1.268	1.344	1.42	1.42	0.076
	Average Transac	1	Createinvoice_select_taxcode	1.88	2.100	2.227	2.217	0.161
	Average Transac	1	CRM_Manage_Role_Clck_Apps_Logon_Links	0	0.087	0.176	0.173	0.087
	Average Transac	1	CRM_Manage_Role_Clck_CRM_HTML_Administ	0.393	0.881	1.383	0.881	0.35
	Average Transac	1	CRM_Manage_Role_Clck_Setup_Home	0.338	0.941	2.15	0.338	0.895
	Average Transac	1	CRM_Manage_Role_Clck_Update	0.343	0.883	1.44	1.367	0.531
	Average Transac	1	CRM_Manage_Role_Clck_User_Maintenance	0.255	0.514	1.128	0.381	0.357
	Average Transac	1	CRM_Manage_Role_Clck_Users	0.18	0.857	1.624	1.441	0.679
	Average Transac	1	CRM_Manage_Role_EBusiness_Links	0.168	0.84	1.588	1.417	0.666
	Average Transac	1	CRM_Manage_Role_Login	0.474	1.043	1.605	1.294	0.436
	Average Transac	1	CRM_Manage_Role_Logout	0.446	0.741	1.098	0.929	0.28
	Average Transac	1	CRM_Manage_Role_Search	0.254	0.649	2.03	0.262	0.395
	Average Transac	1	CRM_Manage_Role_Select_Role	0.181	1.267	3.54	1.154	1.371
	Average Transac	1	CRM_Manage_Role_Select_Test_user	0.179	0.77	1.894	0.817	0.698
	Average Transac	1	DCAP_Receivables_Clck_LineItems_EnterDet	3.495	5.739	10.133	3.589	3.107
	Average Transac	1	DCAP_Receivables_Clck_Transactions	7.079	102.1	209.027	165.122	95.35
	Average Transac	1	DCAP_Receivables_EBusiness_Links	0.172	0.581	1.344	0.227	0.54
	Average Transac	1	DCAP_Receivables_EnterTransDetails_Save	0.407	0.472	0.56	0.449	0.065
	Average Transac	1	DCAP_Receivables_LineItems_Save	0.248	0.293	0.376	0.257	0.068
	Average Transac	1	DCAP_Receivables_Login	0.378	0.736	1.379	0.45	0.456
	Average Transac	1	DCAP_Receivables_LogonLinks	0	0.506	1.011	1.011	0.505
	Average Transac	1	DCAP_Receivables_Logout	0.377	0.539	0.7	0.7	0.162
	Average Transac	1	DCAP_Receivables_ShipTo	1.259	2.085	3.736	1.261	1.167
	Average Transac	1	DCAP_Receivables_Vision_Operation	0.722	0.921	0.989	0.753	0.119
	Average Transac	1	iProcurement_AppsLogonLinks	0	0.341	0.997	0.365	0.497
	Average Transac	1	iProcurement_Choose	1.259	1.662	2.134	1.995	0.405
	Average Transac	1	iProcurement_Clck_AddToCart	1.037	1.055	1.083	1.044	0.02
	Average Transac	1	iProcurement_Clck_AskforSupport	1.395	1.914	2.368	1.68	0.468
	Average Transac	1	iProcurement_Clck_Categories	0.28	0.626	1.3	0.557	0.402
	Average Transac	1	iProcurement_Clck_Ergonomic_Supplies	0.209	0.888	1.226	1.14	0.402
	Average Transac	1	iProcurement_Clck_ViewCart	1.276	1.737	2.37	1.564	0.463
	Average Transac	1	iProcurement_Delete_Item	0.565	0.681	0.865	0.718	0.122
	Average Transac	1	iProcurement_EbusinessHomepage	0.356	0.927	1.711	1.266	0.583
	Average Transac	1	iProcurement_Homepage	0	0.497	1.484	0.504	0.606
	Average Transac	1	iProcurement_Logon	1.172	1.642	2.495	1.46	0.505
	Average Transac	1	iProcurement_Logout	0.934	1.147	1.54	0.967	0.279
	RunningVusers	1	Run	0	4	10	4	3.768

Color	Scale	Measurement	Graph Minimum	Graph Average	Graph Maximum	Graph Median	Graph Std. Deviation
	1	Create_Project_forms_Clck_Apps_Logon_Links	0	0.069	0.279	0.070	0.114
	1	Create_Project_forms_Clck_Copy_To	0.894	1.173	1.607	1.178	0.27
	1	Create_Project_forms_Clck_find	0.942	1.607	2.085	2.081	0.494
	1	Create_Project_forms_clck_open	0.904	1.661	2.115	1.924	0.461
	1	Create_Project_forms_Clck_project_lnk	9.639	32.916	63.78	48.118	23.689
	1	Create_Project_forms_Clck_project_name	0.937	1.062	1.228	1.059	0.105
	1	Create_Project_forms_Clck_project_vision_service	0.616	1.372	2.633	1.165	0.765
	1	Create_Project_forms_EBusiness_Links	0.282	0.706	1.382	0.674	0.414
	1	Create_Project_forms_Login	0.614	1.217	2.844	0.729	0.94
	1	Create_Project_forms_Logout	0.976	1.062	1.094	1.068	0.086
	1	Create_Project_forms_select_Change_Status	1.178	1.307	1.452	1.412	0.126
	1	Create_Project_forms_select_Customer_name	1.463	1.615	1.754	1.723	0.122
	1	Create_Project_forms_select_organization	1.598	1.747	1.966	1.807	0.15
	1	Create_Project_forms_select_PM	1.399	1.659	1.917	1.729	0.19
	1	Createinvoice_Clck_Apps_Logon_Links	0	0.093	0.202	0.169	0.093
	1	Createinvoice_Clck_Distributions	1.662	3.689	6.689	3.345	1.845
	1	Createinvoice_Clck_EBusiness_Links	0.703	0.857	0.962	0.915	0.098
	1	Createinvoice_clck_Entp_Invoices	22.027	36.439	60.566	33.704	14.544
	1	Createinvoice_clck_Payables_Vision_Operations	1.155	1.729	3.126	1.457	0.916
	1	Createinvoice_Login	0.9	1.442	2.37	1.295	0.556
	1	Createinvoice_logout	0.541	0.896	1.057	0.964	0.211
	1	Createinvoice_select_account	2.007	2.461	2.715	2.658	0.278
	1	Createinvoice_select_Save	3.913	4.524	5.116	5.071	0.571
	1	Createinvoice_select_Supplier_Name	0.99	1.344	1.635	1.546	0.26
	1	Createinvoice_select_taxcode	1.88	2.138	2.253	2.217	0.15
	1	CRM_Manage_Role_Clck_Apps_Logon_Links	0	0.087	0.176	0.173	0.087
	1	CRM_Manage_Role_Clck_CRM_HTML_Administ	0.393	0.881	1.383	0.881	0.35
	1	CRM_Manage_Role_Clck_Setup_Home	0.338	1.244	2.19	2.11	0.906
	1	CRM_Manage_Role_Clck_Update	0.343	0.883	1.44	1.367	0.531
	1	CRM_Manage_Role_Clck_User_Maintenance	0.255	0.514	1.128	0.381	0.357
	1	CRM_Manage_Role_Clck_Users	0.181	0.857	1.624	1.441	0.679
	1	CRM_Manage_Role_EBusiness_Links	0.168	0.84	1.588	1.417	0.666
	1	CRM_Manage_Role_Login	0.474	1.043	1.605	1.294	0.436
	1	CRM_Manage_Role_Logout	0.446	0.741	1.098	0.929	0.28
	1	CRM_Manage_Role_Search	0.262	1.146	2.67	1.389	0.993
	1	CRM_Manage_Role_Select_Role	0.181	1.267	3.54	1.154	1.371
	1	CRM_Manage_Role_Select_Test_user	0.179	0.77	1.894	0.817	0.698
	1	DCAP_Receivables_Clck_LineItems_EnterDet	3.495	5.262	10.133	3.62	2.847
	1	DCAP_Receivables_Clck_Transactions	7.079	102.1	209.027	165.122	95.35
	1	DCAP_Receivables_EBusiness_Links	0.172	0.772	1.367	1.32	0.573
	1	DCAP_Receivables_EnterTransDetails_Save	0.407	0.466	0.56	0.457	0.057
	1	DCAP_Receivables_LineItems_Save	0.251	0.295	0.376	0.264	0.052
	1	DCAP_Receivables_Login	0.378	0.892	1.509	1.165	0.507
	1	DCAP_Receivables_LogonLinks	0	0.506	1.103	0.92	0.51
	1	DCAP_Receivables_Logout	0.395	0.539	0.973	0.427	0.252
	1	DCAP_Receivables_ShipTo	1.245	1.879	3.736	1.278	1.072
	1	DCAP_Receivables_Vision_Operation	0.706	0.798	0.989	0.753	0.112
	1	iProcurement_AppsLogonLinks	0	0.341	0.997	0.365	0.407
	1	iProcurement_Choose	1.261	1.663	2.134	1.995	0.405
	1	iProcurement_Clck_AddToCart	0.803	1.051	1.272	1.083	0.167
	1	iProcurement_Clck_AskforSupport	1.107	1.71	2.368	1.682	0.446
	1	iProcurement_Clck_Categories	0.28	0.626	1.3	0.557	0.402
	1	iProcurement_Clck_Ergonomic_Supplies	0.209	0.888	1.226	1.14	0.402
	1	iProcurement_Clck_ViewCart	0.513	1.621	2.37	2.039	0.701
	1	iProcurement_Delete_Item	0.565	0.681	0.865	0.718	0.122
	1	iProcurement_EbusinessHomepage	0.356	0.927	1.711	1.266	0.583
	1	iProcurement_Homepage	0	0.497	1.484	0.504	0.606
	1	iProcurement_Logon	1.172	1.642	2.495	1.46	0.505
	1	iProcurement_Logout	0.906	1.095	1.54	0.967	0.258

Step 7 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
```

Expected Results

- We expect WAAS to accelerate transaction response times.
- We expect transaction response times to be slightly higher in comparison to DCa.
- We expect Oracle transactions submitted through Load Runner will 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.

Results

Oracle Apps Traffic from Branch 3 to DCb with WAAS passed.

Global E-Business Suite Across Data Centers

Global E-Business Suite Across Data Centers validates the distribution of Application traffic across both Data Centers A and B. Application traffic generated from 3 branch clients is sent to both Data Centers and GSS used weighted round robin mode to distribute the connections.

The following tests were performed:

- Global Distribution of Oracle Apps Traffic without WAAS, page 8-48
- Global Distribution of Oracle Apps Traffic with WAAS, page 8-50

Global Distribution of Oracle Apps 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.

Test Procedure

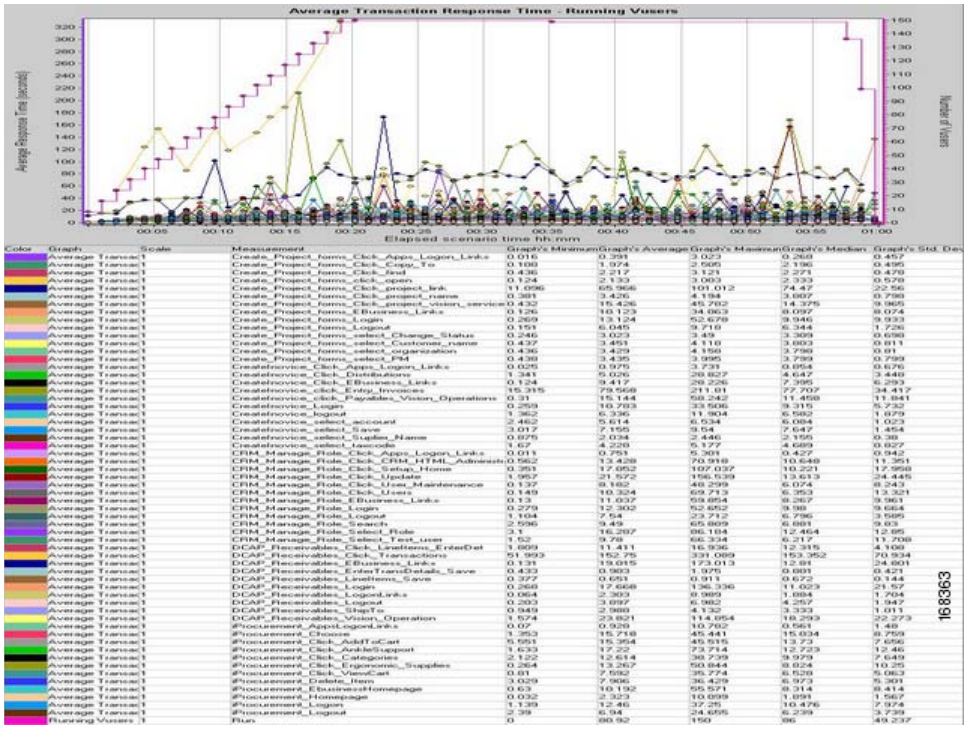
The procedure used to perform the Global Distribution of Oracle Apps Traffic without WAAS test follows:

Step 1 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 datacenters.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool.

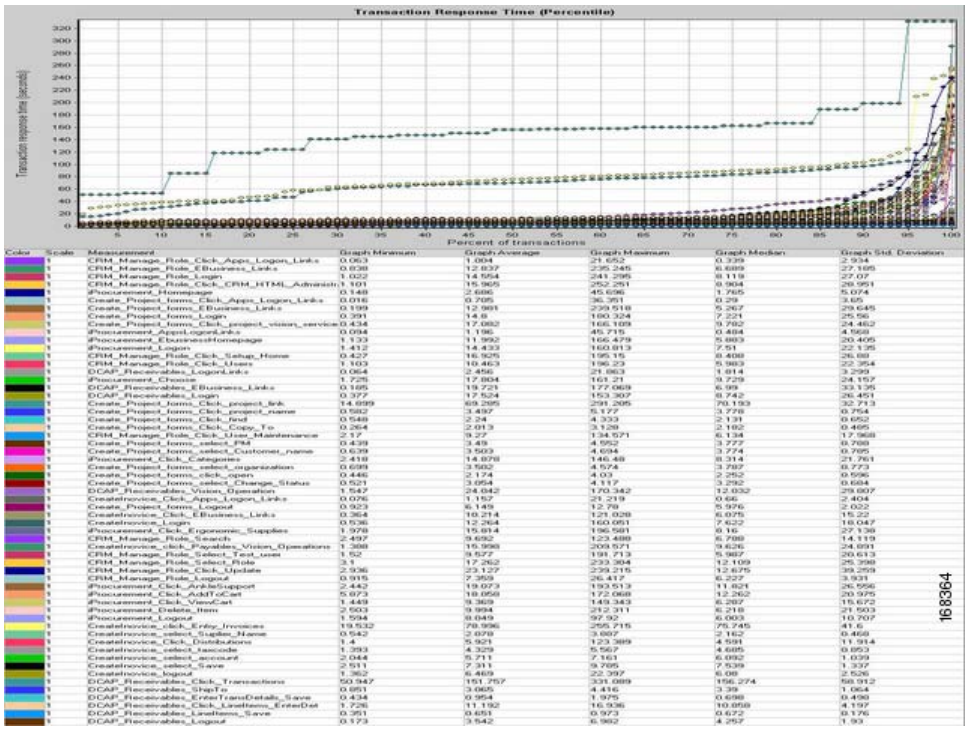
Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

Step 3 Initiate the Load Runner generated traffic which will run for approximately 1 hour.

Step 4 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays 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.



Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Dev.
	Average Transac	1	Create_Project_forms_Clck_Apps_Logon_Links	0.016	0.391	3.023	0.269	0.457
	Average Transac	1	Create_Project_forms_Clck_Copy_To	0.188	1.974	2.595	2.196	0.495
	Average Transac	1	Create_Project_forms_Clck_find	0.436	2.217	3.121	2.271	0.479
	Average Transac	1	Create_Project_forms_clck_open	0.124	2.133	3.063	2.333	0.579
	Average Transac	1	Create_Project_forms_Clck_project_lnk	11.096	65.966	101.012	74.47	22.56
	Average Transac	1	Create_Project_forms_Clck_project_name	0.381	3.426	4.184	3.807	0.798
	Average Transac	1	Create_Project_forms_Clck_project_vision_service	0.432	15.426	45.762	14.375	9.965
	Average Transac	1	Create_Project_forms_EBusiness_Links	0.126	10.123	34.063	8.097	8.074
	Average Transac	1	Create_Project_forms_Login	0.269	13.124	52.678	9.946	9.933
	Average Transac	1	Create_Project_forms_Logout	0.151	6.045	9.716	6.344	1.726
	Average Transac	1	Create_Project_forms_select_Change_Status	0.246	3.023	3.49	3.309	0.698
	Average Transac	1	Create_Project_forms_select_Customer_name	0.437	3.451	4.116	3.803	0.811
	Average Transac	1	Create_Project_forms_select_organization	0.436	3.429	4.156	3.798	0.81
	Average Transac	1	Create_Project_forms_select_PM	0.436	3.435	3.996	3.798	0.798
	Average Transac	1	CreateInvoice_Clck_Apps_Logon_Links	0.025	0.975	3.731	0.954	0.676
	Average Transac	1	CreateInvoice_Clck_Distributions	1.341	5.026	28.827	4.647	3.449
	Average Transac	1	CreateInvoice_Clck_EBusiness_Links	0.124	9.417	28.226	7.395	6.293
	Average Transac	1	CreateInvoice_clck_Entg_Invoices	15.315	79.568	211.81	77.707	34.417
	Average Transac	1	CreateInvoice_clck_Payables_Vision_Operations	0.31	15.144	58.242	11.458	11.841
	Average Transac	1	CreateInvoice_Login	0.258	10.783	33.506	9.315	5.732
	Average Transac	1	CreateInvoice_logout	1.362	6.336	11.904	6.582	1.679
	Average Transac	1	CreateInvoice_select_account	2.462	5.614	6.534	6.084	1.023
	Average Transac	1	CreateInvoice_select_Save	3.017	7.155	9.54	7.647	1.454
	Average Transac	1	CreateInvoice_select_Supplier_Name	0.875	2.034	2.446	2.155	0.38
	Average Transac	1	CreateInvoice_select_taxcode	1.67	4.229	5.177	4.689	0.627
	Average Transac	1	CRM_Manage_Role_Clck_Apps_Logon_Links	0.011	0.751	5.301	0.427	0.942
	Average Transac	1	CRM_Manage_Role_Clck_CRM_HTML_Admnsrtn	0.562	13.428	70.918	10.648	11.351
	Average Transac	1	CRM_Manage_Role_Clck_Setup_Home	0.351	17.652	107.037	10.221	17.958
	Average Transac	1	CRM_Manage_Role_Clck_Update	1.957	21.572	116.539	13.613	24.445
	Average Transac	1	CRM_Manage_Role_Clck_User_Maintenance	0.137	8.162	48.299	6.074	8.243
	Average Transac	1	CRM_Manage_Role_Clck_Users	0.149	10.324	69.713	6.353	13.321
	Average Transac	1	CRM_Manage_Role_EBusiness_Links	0.13	11.037	59.654	8.267	9.961
	Average Transac	1	CRM_Manage_Role_Login	0.279	12.302	52.652	9.98	9.664
	Average Transac	1	CRM_Manage_Role_Logout	1.104	7.54	23.712	6.796	3.595
	Average Transac	1	CRM_Manage_Role_Search	2.596	9.49	65.869	6.881	9.03
	Average Transac	1	CRM_Manage_Role_Select_Role	3.1	16.267	86.184	12.464	12.95
	Average Transac	1	CRM_Manage_Role_Select_Test_user	1.52	9.78	66.334	6.217	11.708
	Average Transac	1	DCAP_Receivables_Clck_LineItems_EnterDat	1.869	11.411	16.936	12.315	4.108
	Average Transac	1	DCAP_Receivables_Clck_Transactions	51.993	152.75	331.089	153.352	70.934
	Average Transac	1	DCAP_Receivables_EBusiness_Links	0.131	19.015	173.013	12.81	24.601
	Average Transac	1	DCAP_Receivables_EnterTranDetails_Save	0.433	0.903	1.975	0.881	0.421
	Average Transac	1	DCAP_Receivables_LineItems_Save	0.377	0.651	0.911	0.672	0.144
	Average Transac	1	DCAP_Receivables_Login	0.268	17.668	136.336	11.623	21.57
	Average Transac	1	DCAP_Receivables_LogonLinks	0.064	2.303	8.989	1.884	1.704
	Average Transac	1	DCAP_Receivables_Logout	0.293	3.897	6.982	4.257	1.947
	Average Transac	1	DCAP_Receivables_ShipTo	0.949	2.989	4.132	3.333	1.011
	Average Transac	1	DCAP_Receivables_Vision_Operation	1.574	23.821	114.854	18.293	22.273
	Average Transac	1	#ocurement_AppsLogonLinks	0.07	0.928	10.762	0.561	1.48
	Average Transac	1	#ocurement_Choose	1.353	15.718	45.441	15.034	8.759
	Average Transac	1	#ocurement_Clck_AddToCart	5.551	15.354	45.515	13.73	7.656
	Average Transac	1	#ocurement_Clck_AnkleSupport	1.633	17.22	73.714	12.723	12.46
	Average Transac	1	#ocurement_Clck_Categories	2.122	12.614	38.739	9.979	7.649
	Average Transac	1	#ocurement_Clck_Ergonomic_Supplies	0.264	13.267	50.844	8.024	10.25
	Average Transac	1	#ocurement_Clck_ViewCart	0.81	7.592	35.774	6.528	5.063
	Average Transac	1	#ocurement_Delete_Item	3.029	7.906	36.429	6.973	5.301
	Average Transac	1	#ocurement_EbusinessHomepage	0.63	10.192	55.571	8.314	8.414
	Average Transac	1	#ocurement_Homepage	0.032	2.323	10.699	1.891	1.567
	Average Transac	1	#ocurement_Logon	1.139	12.46	37.25	10.476	7.974
	Average Transac	1	#ocurement_Logout	2.39	6.94	24.655	6.239	3.739
	Running Vusers	1	Run	0	80.92	150	86	49.237



Expected Results

- 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 the majority of Oracle transactions submitted through Load Runner from all branches across both data centers to be completed successfully.
- We expect that, without the acceleration capabilities provided by WAAS, there to be time out of incoming requests from remote branch offices.
- We expect the GSS to direct the Application traffic to both data centers as per design criteria.
- We expect the CSM to load balance the connections across the application hosts in their respective data centers.

Results

Global Distribution of Oracle Apps Traffic without WAAS passed.

Global Distribution of Oracle Apps 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 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 Apps Traffic with WAAS test follows:

Step 1 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 datacenters.To simulate connections from multiple IP addresses, leverage function of IP spoofing from LR tool.

Step 2 On the GSS, verify that the DNS rule wwwin-oefin is configured properly by issuing the **show tech-support config** command.

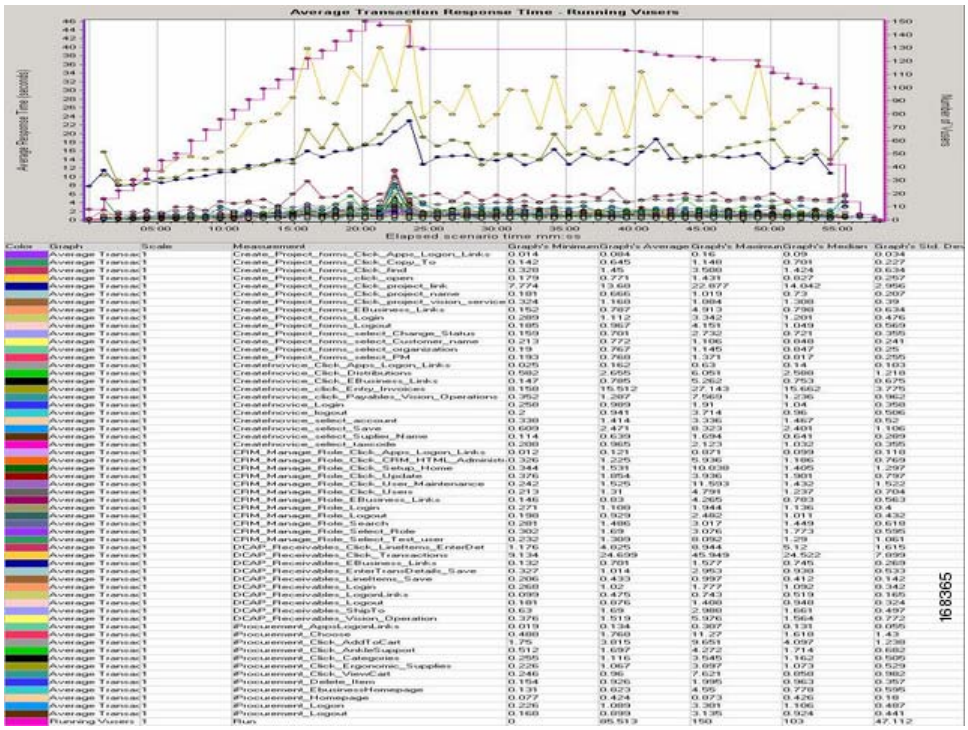
Step 3 Clear the appropriate counters on the WAAS devices by issuing the **clear statistics all** command.

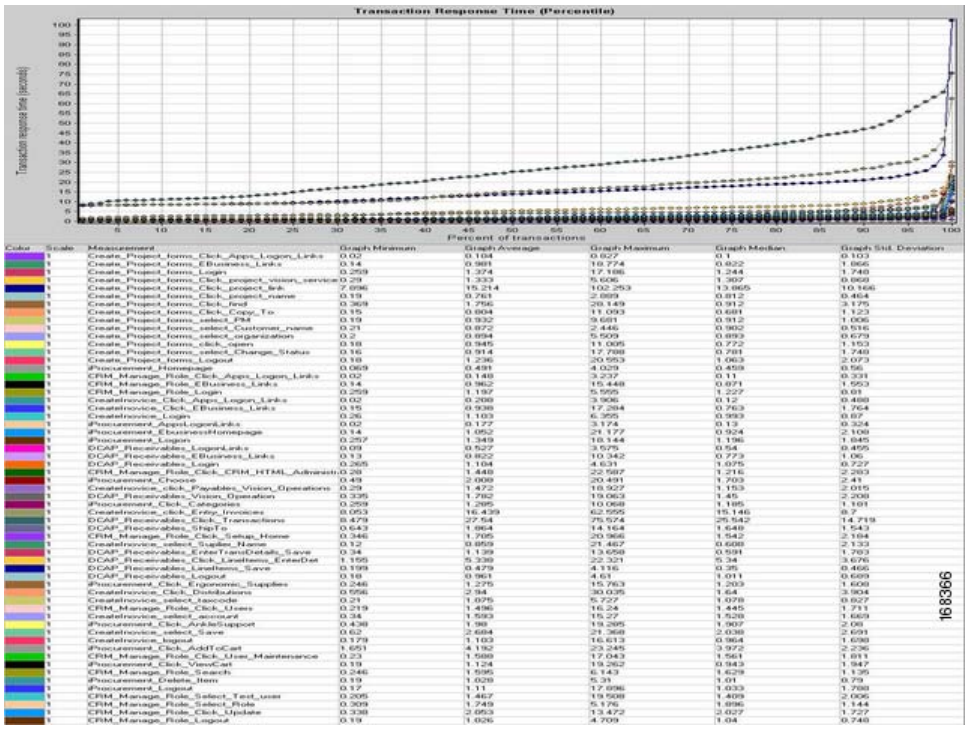
Step 4 Initiate the Load Runner generated traffic which will run for approximately 1 hour.

Step 5 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 6 Once the traffic has completed save the results on the Load Runner controller. LR tool will generate 2 graphs. The first graph displays 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 7 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
```

Expected Results

- We expect WAAS to 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 the GSS to direct the Application traffic to both data centers as per design criteria.
- We expect the CSM to load balance the connections across the application hosts in their respective data centers.

Results

Global Distribution of Oracle Apps Traffic with WAAS passed.

CHAPTER 9

Microsoft Exchange 2003

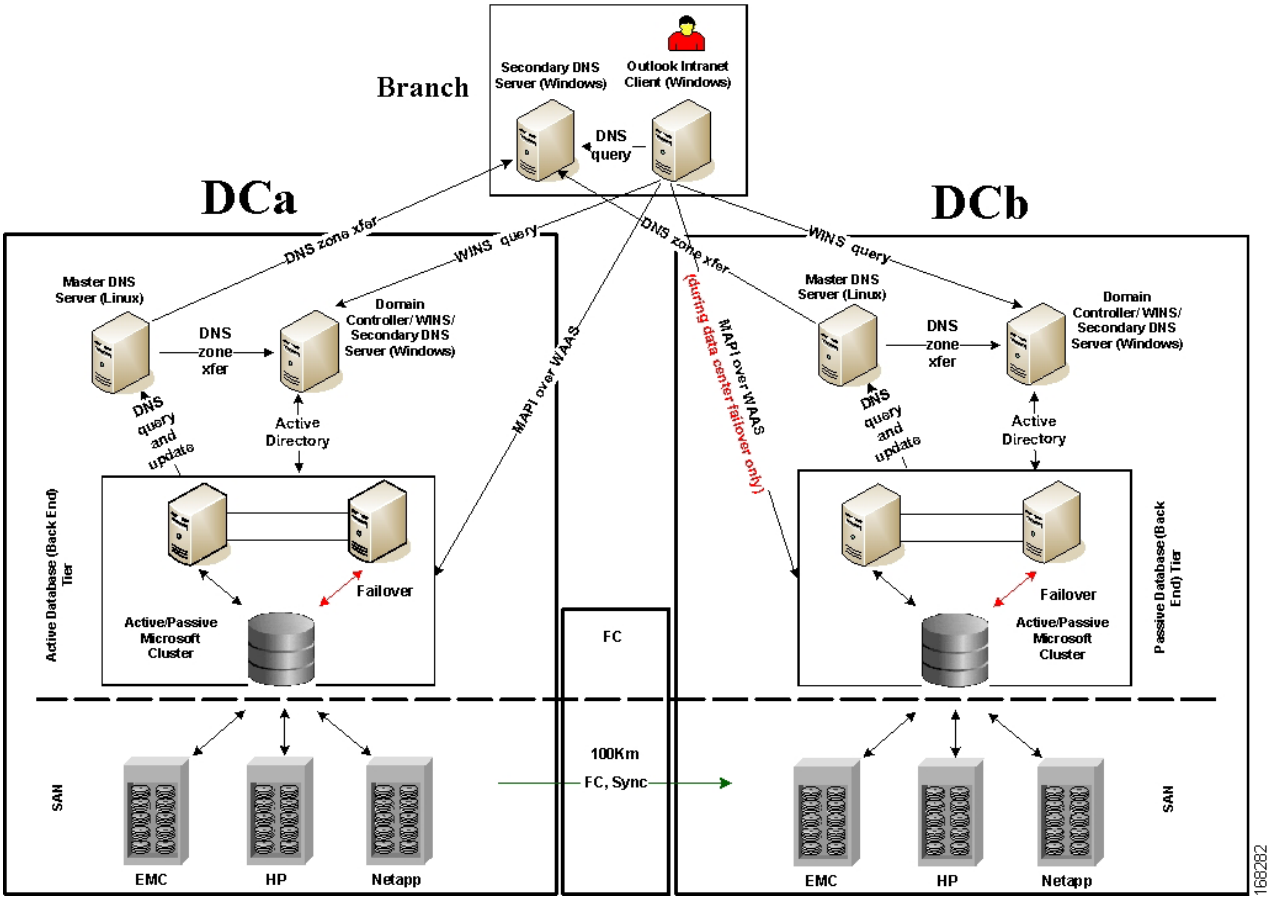
DCAP 3.0 testing includes Microsoft Exchange 2003 and Outlook 2003.

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 fiber 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 LoadSim and 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. The DCAP roadmap tentatively includes Exchange 2007, asynchronous replication, and other features such as front end servers, server load balancing, firewall, SSL offload, and proxy.

Exchange Topology

Figure 9-1 depicts the primary Microsoft Exchange 2003 application components and some basic data flow information to show how they relate to each other.

Figure 9-1 DCAP Exchange Test Topology



The components include the following:

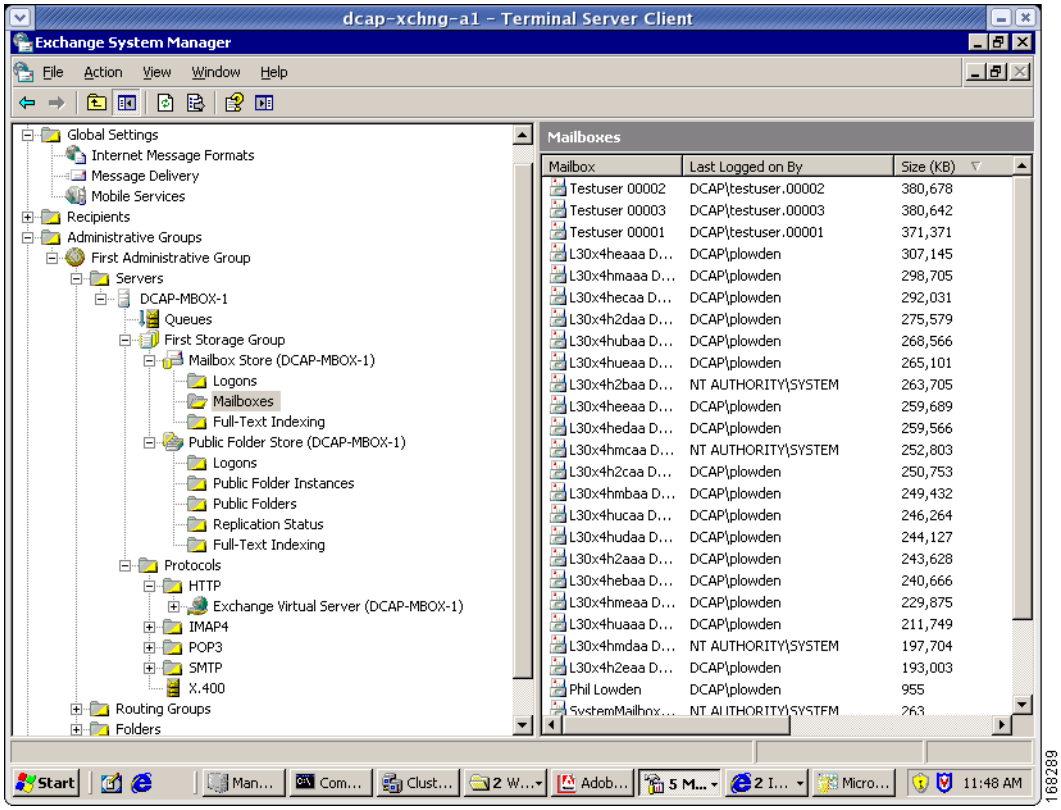
- A primary active/passive Microsoft Windows cluster in DCa and a similarly configured failover active/passive cluster in DCb. These clusters provide a high-availability environment for the Microsoft Exchange 2003 virtual server called "DCAP-MBOX-1" that is used throughout the tests. These clusters are separate and self-contained; that is, the hosts are not configured as part of a geographically dispersed cluster.
- SAN storage 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 data center A to data center B. Fiber channel-attached storage from three different vendors, EMC, Hewlett Packard, and Network Appliance, is used, and the replication mechanism is SRDF/S, Continuous Access XP Synchronous, and synchronous SnapMirror, respectively.
- 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, WAN, WAAS, and LAN topologies used by Exchange, please see the corresponding sections in this document.

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 9-2 shows a screen dump of Exchange System Manager. This is an image capture of the Microsoft Exchange System Manager application running on the primary DCAP 3.0 Exchange cluster. System Manager is used to manage an Exchange environment.

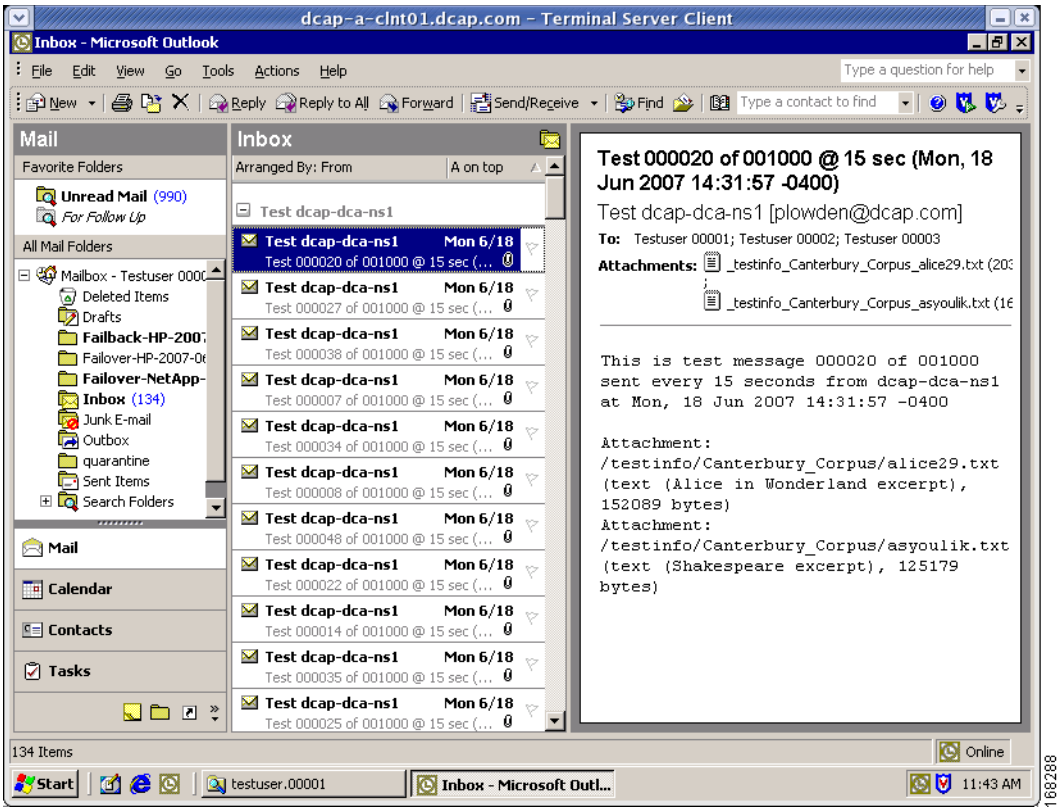
Figure 9-2 DCAP MS Exchange 2003 Exchange System Manager Screen Capture



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 9-3 shows a screen dump of Outlook 2003. This is an image capture of the Microsoft Outlook 2003 application running on a DCAP 3.0 Exchange client. Outlook is used to access email and scheduling information Exchange environment.

Figure 9-3 Outlook 2003 Screen Capture



From a storage perspective, the DCAP SAN testbed and storage frames from EMC, HP, and Network Appliance, 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 9-4 shows a screen dump of Cluster Administrator from the primary cluster node depicting the cluster resource group.This is an image capture of the Cluster Administrator application on the Microsoft Cluster server used as the primary DCAP 3.0 Exchange cluster showing the main cluster resource group.

Figure 9-4 Cluster Administrator Cluster Resource Group Screen Capture

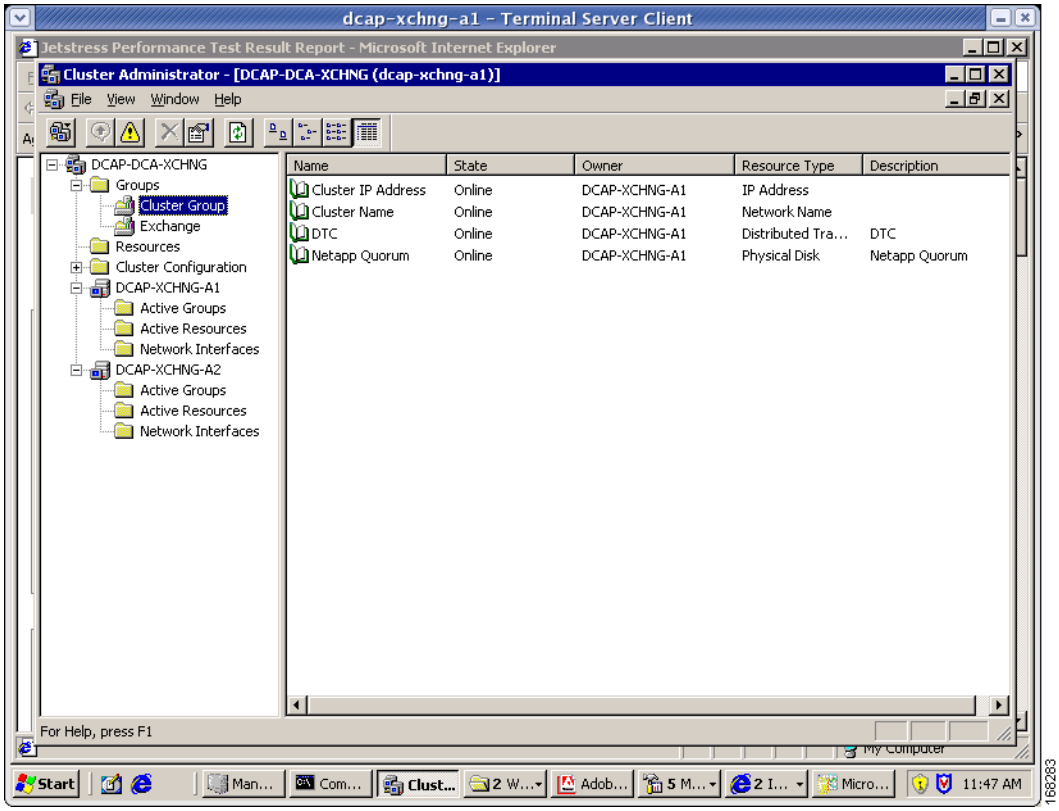
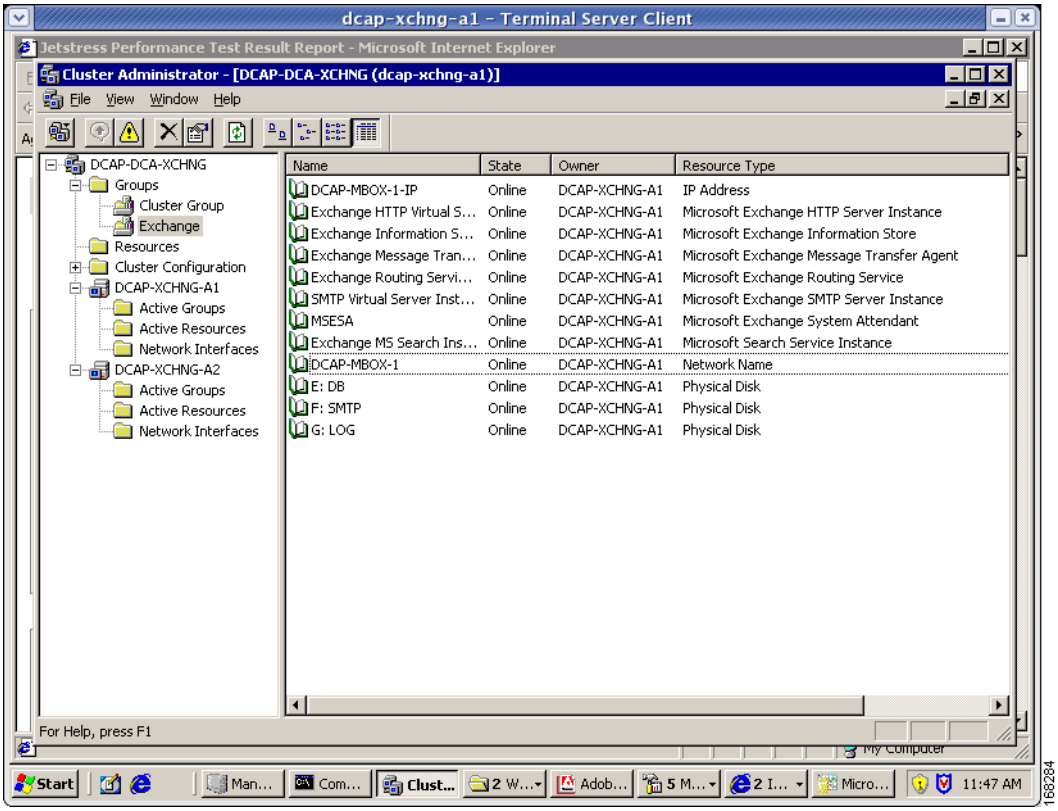


Figure 9-5 shows a screen dump of Cluster Administrator from the primary cluster node depicting the Exchange resource group. This is an image capture of the Cluster Administrator application on the Microsoft Cluster server used as the primary DCAP 3.0 Exchange cluster showing the Exchange Virtual Server resource group.

Figure 9-5 Cluster Administrator Exchange Service Group Screen Capture



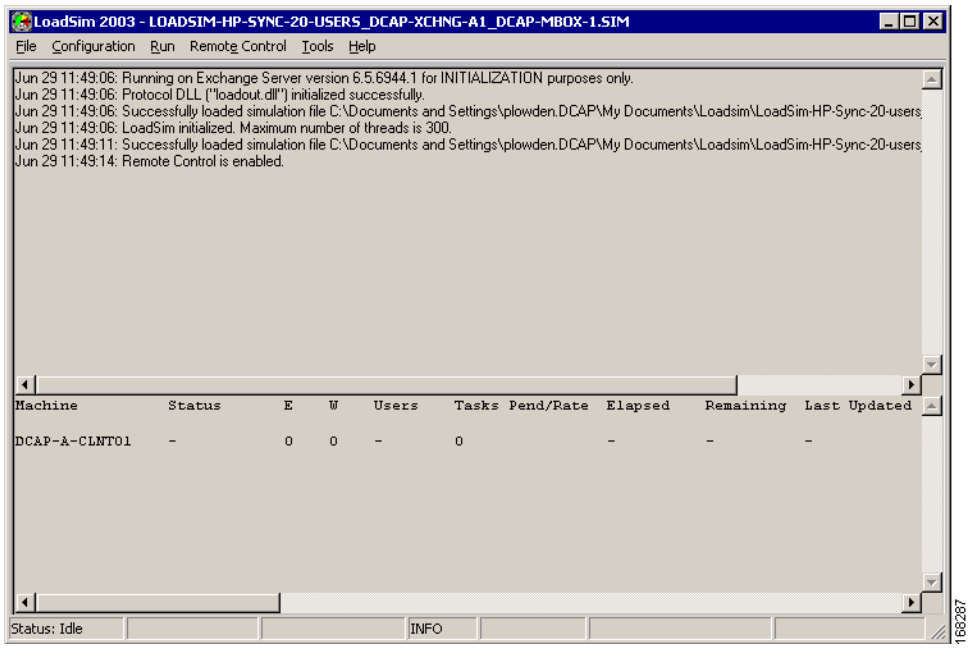
The Cisco DCAP 3.0 Exchange tests fall into two major categories: Fabric Extension and Disaster Recovery.

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.

Two Microsoft tools facilitated Fabric Extension testing. These included LoadSim for simulating the performance load of MAPI clients and Jetstress for simulating Exchange disk I/O load. For more information on LoadSim and Jetstress, see http://go.microsoft.com/fwlink/?linkid=27882 and http://go.microsoft.com/fwlink/?linkid=27883.

LoadSim was used to simulate the activity of 20 Exchange clients while the SAN replication path was configured without FC write acceleration. 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 LoadSim tests for EMC and HP storage, but because NetApp replication uses the FC-VI protocol rather than FCP, write acceleration had no effect on it. Please see the test results for more information. Figure 9-6 shows a screen dump of LoadSim from the primary cluster node. This is an image capture of the Microsoft Loadsim application running on on the primary DCAP 3.0 Exchange cluster. Loadsim is used to stress test an Exchange environment.

Figure 9-6 LoadSim Screen Capture



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 9-1 summarizes these Jetstress results.

Table 9-1 Jetstress Test Results

Vendor	Without FCWA		With FCWA		Write Speedup
	Write/Sec	Total OPS	Write/Sec	Total IOPS	
EMC	427	595	646	944	51%
HP	817	1417	877	1441	7%
NetApp	1777	3365	1689	3242	-5%

Please see detailed test results for more information.

Figure 9-7 shows a screen dump of Jetstress from the primary cluster node after configuration and before execution of a test. This is an image capture of the Microsoft Jetstress application running on the primary DCAP 3.0 Exchange cluster. Jetstress is used to stress test storage resources.

Figure 9-7 Jetstress Screen Capture

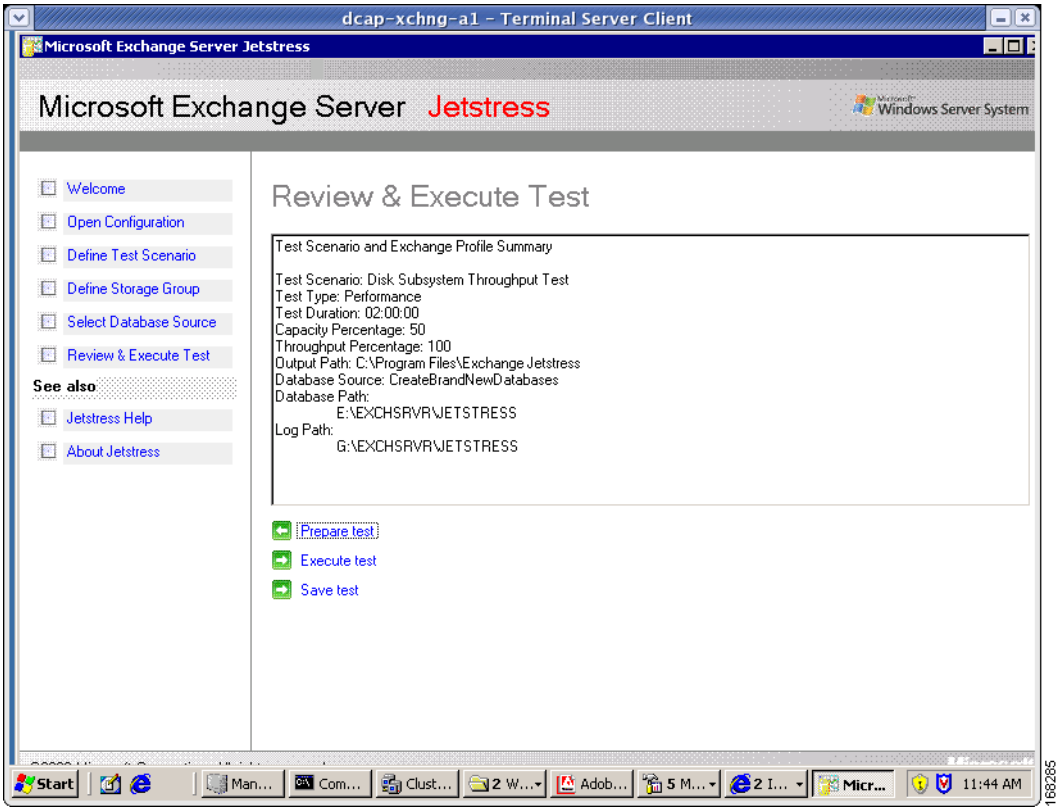
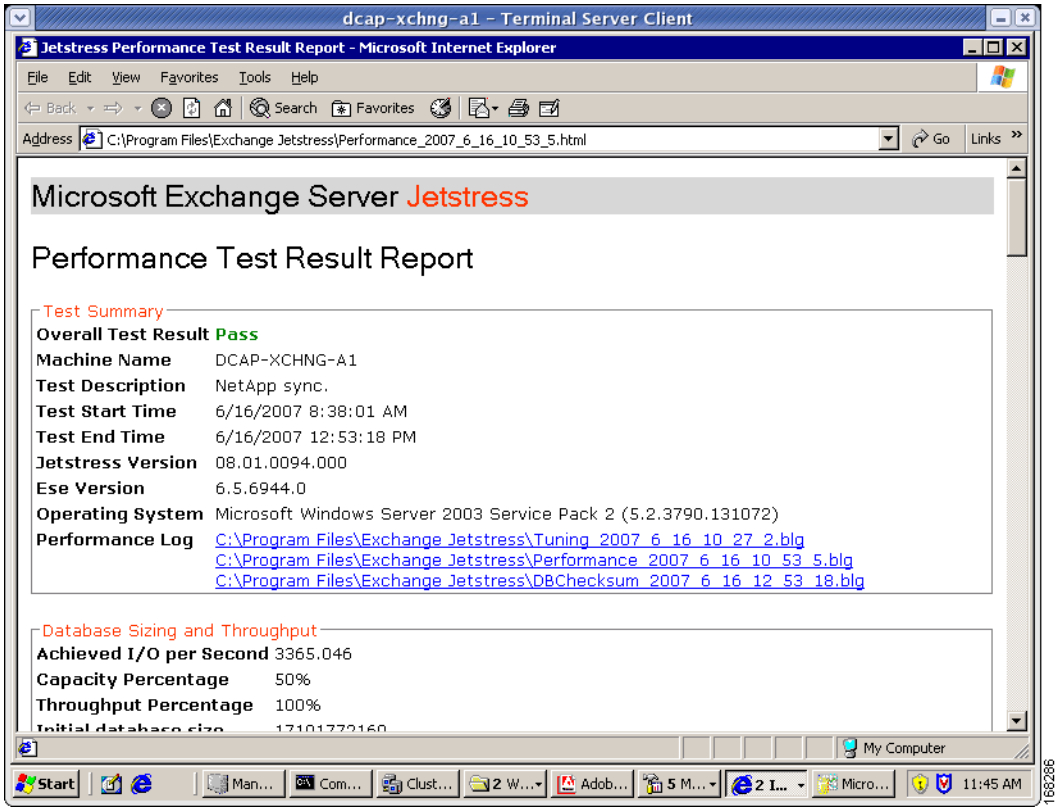


Figure 9-8 shows a screen dump of a Jetstress HTML report from the primary cluster node after execution of a test. This is an image capture of an HTML report from a run of the Microsoft Jetstress application on the primary DCAP 3.0 Exchange cluster. Jetstress is used to stress test storage resources.

Figure 9-8 Jetstress Report Screen Capture



Note that 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).

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 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 were performed with fiber 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 data center B. 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 included updating DNS at the branches to allow clients to adjust to the change in the IP address. (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 fail over 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 failed over to data center B 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 fail over 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 data center A were received by clients once Exchange was online in data center B.

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 data center B. By the end of the test, Exchange was once again running on the primary cluster in data center A and Outlook clients at each branch were checked to make sure that no email was lost.

Table 9-2 summarizes the results of failover and failback testing.

Table 9-2 Failover and Failback Test Results

Vendor	Failover		Failback	
	RPO	RTO	RPO	RTO
EMC	0	23 min	0	15 min
HP	0	20 min	0	12 min
NetApp	0	17min	0	11 min

For details about how to perform the failover and failback, refer to the "Disaster Recovery" section on page 10-1.

MS Exchange 2003 Test Results Summary

Table 9-3 summarizes tests executed as part of the Cisco DCAP 3.0 testing initiative. Table 9-3 includes the feature or function tested, the section that describes the feature set the feature or function belongs to, the component tests for each feature or function, and whether the test is new in this phase of DCAP testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested. DCAP is designed to cover critical path areas and augment ongoing regression and systems testing.

Table 9-3 Cisco DCAP 3.0 Disaster Recovery Testing Summary

Test Suites	Features/Functions	Tests	Results
Fabric Extension	EMC, page 9-12	1. Jetstress with EMC Sync Replication (100km with FC Write Acceleration) 2. Jetstress with EMC Sync Replication (100km no FC Write Acceleration) 3. LoadSim-EMC-Sync-100km-FC WA 4. LoadSim-EMC-Sync-100km-no FC WA	
	NetApp, page 9-16	1. Jetstress-NetApp-Sync-100km-FC WA 2. Jetstress-NetApp-Sync-100km-no FC WA 3. LoadSim-NetApp-Sync-100km-FC WA 4. LoadSim-NetApp-Sync-100km-no FC WA	
	HP, page 9-19	1. Jetstress-HP-Sync-100km-FC WA 2. Jetstress-HP-Sync-100km-no FC WA 3. LoadSim-HP-Sync-100km-FC WA 4. LoadSim-HP-Sync-100km-no FC WA	
Disaster Recovery	Fail Over, page 9-24	1. Exchange-EMC-Fail-Back-Sync-100km-WA 2. Exchange-NetApp-Fail-Back-Sync-100km-WA 3. Exchange-HP-Fail-Back-Sync-100km-WA	
	Fail Back, page 9-28	1. Exchange-EMC-Fail-Over-Sync-100km-WA 2. Exchange-NetApp-Fail-Over-Sync-100km-WA 3. Exchange-HP-Fail-Over-Sync-100km-WA	

MS Exchange 2003 Test Cases

Functionality critical to global enterprises inCisco DCAP 3.0 MS Exchange testing is described in the following sections. Refer to Cisco Data Center Assurance Program (DCAP) 3.0 Configurations document for test device configurations.

- Fabric Extension, page 9-11
- Disaster Recovery, page 9-24

Fabric Extension

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 and Loadsim tools are used.

The following test features were conducted:

- EMC, page 9-12

- NetApp, page 9-16
- HP, page 9-19

EMC

The synchronous replication tests for EMC tests SRDF/S over FC with and without FCIP write acceleration.

The following test was performed:

- Jetstress with EMC Sync Replication (100km with FC Write Acceleration), page 9-12
- Jetstress with EMC Sync Replication (100km no FC Write Acceleration), page 9-13
- LoadSim-EMC-Sync-100km-FC WA, page 9-14
- LoadSim-EMC-Sync-100km-no FC WA, page 9-15

Jetstress with EMC Sync Replication (100km with 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 datacenter 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. 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

- 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.

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 datacenter 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. 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

- 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.
- We expect that there will be no unacceptable impact on the CPU or memory of the devices under test.

Results

Jetstress with EMC Sync Replication (100km no FC Write Acceleration) passed.

LoadSim-EMC-Sync-100km-FC WA

This test runs Microsoft's 1-hour LoadSim utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. LoadSim allows one or more hosts to logon to Exchange and create transactions typical of an average email and public folder user. Together with Windows performance monitor, LoadSim verifies the performance of the server as well as the disk subsystem, including the SAN connectivity and replication infrastructure. For more information on Jetstress, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27882. 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 datacenter 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 LoadSim-EMC-Sync-100km-FC WA test follows:

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 LoadSim with the primary Exchange cluster node as a controller node and a client Windows server as a remote controlled 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 20 users. User type should be MMB3. Use defaults for Distribution Lists and Public Folders. Use Stress Mode (Max. Speed).

Step 4 After LoadSim is done, check LoadSim log files, the Windows performance monitor logs, 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

- We expect that a one-hour run of Loadsim will finish successfully and show better performance with FC Write Acceleration enabled than without it.

- We expect no CPU or memory problems.

Results

LoadSim-EMC-Sync-100km-FC WA passed.

LoadSim-EMC-Sync-100km-no FC WA

This test runs Microsoft's 1-hour LoadSim utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. LoadSim allows one or more hosts to logon to Exchange and create transactions typical of an average email and public folder user. Together with Windows performance monitor, LoadSim verifies the performance of the server as well as the disk subsystem, including the SAN connectivity and replication infrastructure. For more information on LoadSim, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27882. 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 datacenter separated by 100 km (1 ms round-trip time) of latency. The mechanism used is EMC SRDF/S. FC write acceleration is not enabled on the replication MDS switches.

Test Procedure

The procedure used to perform the LoadSim-EMC-Sync-100km-no FC WA test follows:

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 host performance data, interface counters, host status, and replication status throughout the test.

Step 3 Run LoadSim with the primary Exchange cluster node as a controller node and a client Windows server as a remote controlled 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 20 users. User type should be MMB3. Use defaults for Distribution Lists and Public Folders. Use Stress Mode (Max. Speed).

Step 4 After LoadSim is done, check LoadSim log files, the Windows performance monitor logs, 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

- We expect that a one-hour run of Loadsim will finish successfully and show better performance with FC Write Acceleration enabled than without it.
- We expect no CPU or memory problems.

Results

LoadSim-EMC-Sync-100km-no FC WA passed.

NetApp

The synchronous replication test for NetApp tests synchronous SnapMirror with and without FC write acceleration.

The following tests were performed:

- Jetstress-NetApp-Sync-100km-FC WA, page 9-16
- Jetstress-NetApp-Sync-100km-no FC WA, page 9-17
- LoadSim-NetApp-Sync-100km-FC WA, page 9-18
- LoadSim-NetApp-Sync-100km-no FC WA, page 9-19

Jetstress-NetApp-Sync-100km-FC WA

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 datacenter 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 WA test follows:

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

- 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 WA passed.

Jetstress-NetApp-Sync-100km-no FC WA

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 datacenter 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

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 edisbled. 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

- 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-no FC WA passed.

LoadSim-NetApp-Sync-100km-FC WA

This test runs Microsoft's 1-hour LoadSim utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. LoadSim allows one or more hosts to logon to Exchange and create transactions typical of an average email and public folder user. Together with Windows performance monitor, LoadSim verifies the performance of the server as well as the disk subsystem, including the SAN connectivity and replication infrastructure. For more information on LoadSim, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27882. 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 datacenter 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 LoadSim-NetApp-Sync-100km-FC WA test follows:

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 host performance data, interface counters, host status, and replication status throughout the test.

Step 3 Run LoadSim with the primary Exchange cluster node as a controller node and a client Windows server as a remote controlled 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 20 users. User type should be MMB3. Use defaults for Distribution Lists and Public Folders. Use Stress Mode (Max. Speed).

Step 4 After LoadSim is done, check LoadSim log files, the Windows performance monitor logs, 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

- We expect that a one-hour run of Loadsim will finish successfully 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

LoadSim-NetApp-Sync-100km-FC WA passed.

LoadSim-NetApp-Sync-100km-no FC WA

This test runs Microsoft's 1-hour LoadSim utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. LoadSim allows one or more hosts to logon to Exchange and create transactions typical of an average email and public folder user. Together with Windows performance monitor, LoadSim verifies the performance of the server as well as the disk subsystem, including the SAN connectivity and replication infrastructure. For more information on LoadSim, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27882. 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 datacenter 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 LoadSim-NetApp-Sync-100km-no FC WA test follows:

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 host performance data, interface counters, host status, and replication status throughout the test.

Step 3 Run LoadSim with the primary Exchange cluster node as a controller node and a client Windows server as a remote controlled 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 20 users. User type should be MMB3. Use defaults for Distribution Lists and Public Folders. Use Stress Mode (Max. Speed).

Step 4 After LoadSim is done, check LoadSim log files, the Windows performance monitor logs, 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

- We expect that a one-hour run of Loadsim will finish successfully 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

LoadSim-NetApp-Sync-100km-no FC WA passed.

HP

The synchronous replication test for HP tests HP Continuous Access XP Synchronous replication with and without FC write acceleration.

The following tests were performed:

- Jetstress-HP-Sync-100km-FC WA, page 9-20
- Jetstress-HP-Sync-100km-no FC WA, page 9-21
- LoadSim-HP-Sync-100km-FC WA, page 9-22
- LoadSim-HP-Sync-100km-no FC WA, page 9-22

Jetstress-HP-Sync-100km-FC WA

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 datacenter 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 WA test follows:

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

- 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 WA passed.

Jetstress-HP-Sync-100km-no FC WA

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 datacenter 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-no FC WA test follows:

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

- 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-no FC WA passed.

LoadSim-HP-Sync-100km-FC WA

This test runs Microsoft's 1-hour LoadSim utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. LoadSim allows one or more hosts to logon to Exchange and create transactions typical of an average email and public folder user. Together with Windows performance monitor, LoadSim verifies the performance of the server as well as the disk subsystem, including the SAN connectivity and replication infrastructure. For more information on Jetstress, including download instructions, see http://go.microsoft.com/fwlink/?linkid=27882. 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 datacenter 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 LoadSim-HP-Sync-100km-FC WA test follows:

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 host performance data, interface counters, host status, and replication status throughout the test.

Step 3 Run LoadSim on primary Exchange cluster node against 2 remote nodes 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 20 users per node. User type should be MMB3. Use defaults for Distribution Lists and Public Folders. Use Stress Mode (Max. Speed).

Step 4 After LoadSim is done, check LoadSim log files, the Windows performance monitor logs, 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

- We expect that a one-hour run of Loadsim will finish successfully and show better performance with FC Write Acceleration enabled than without it.
- We expect no CPU or memory problems.

Results

LoadSim-HP-Sync-100km-FC WA passed.

LoadSim-HP-Sync-100km-no FC WA

This test runs Microsoft's 1-hour LoadSim utility (32 bit version) on an Exchange database that's replicated to a remote data center for disaster recovery/business continuance purposes. LoadSim allows one or more hosts to logon to Exchange and create transactions typical of an average email and public folder user. Together with Windows performance monitor, LoadSim verifies the performance of the server as well as the disk subsystem, including the SAN connectivity and replication infrastructure. For more information on LoadSim, including download instructions, see

http://go.microsoft.com/fwlink/?linkid=27882. 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 datacenter separated by 100 km (1 ms round-trip time) of latency. The mechanism used is HP Continuous Access XP Sync. FC write acceleration is not enabled on the replication MDS switches.

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 LoadSim-HP-Sync-100km-no FC WA test follows:

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 host performance data, interface counters, host status, and replication status throughout the test.

Step 3 Run LoadSim with the primary Exchange cluster node as a controller node and a client Windows server as a remote controlled 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 20 users. User type should be MMB3. Use defaults for Distribution Lists and Public Folders. Use Stress Mode (Max. Speed).

Step 4 After LoadSim is done, check LoadSim log files, the Windows performance monitor logs, 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

- We expect that a one-hour run of Loadsim will finish successfully and show better performance without FC Write Acceleration enabled than with it.
- We expect the switch to Forward on VLAN 1133 to one aggregation switch and Blocking on another.
- We expect the switch to 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

LoadSim-HP-Sync-100km-no FC WA passed.

Disaster Recovery

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 fiber channel write acceleration enabled on the transit fabric.

The following test features were conducted:

- Fail Over, page 9-24
- Fail Over, page 9-24

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.

The following tests were performed:

- Exchange-EMC-Fail-Back-Sync-100km-WA, page 9-24
- Exchange-NetApp-Fail-Back-Sync-100km-WA, page 9-25
- Exchange-HP-Fail-Back-Sync-100km-WA, page 9-27

Exchange-EMC-Fail-Back-Sync-100km-WA

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 datacenter 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-WA test follows:

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 not enabled, the correct latency is applied, Exchange is running on the primary node in the primary cluster, 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 back Exchange, including the storage and cluster: 0. Using Exchange System Manager, dismount the mailbox and public Folders stores on the primary fail over cluster node.

1. 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).
2. On a host with Solutions Enabler, issue a "symrdf failback" command for the appropriate RDF group and device list. Rescan disks as needed using Disk Manager on the primary cluster node.
3. On the primary cluster node, create or update a network name resource that depends on an IP address resource appropriate for the primary data center; set the network name resource to *not* use Kerberos or ensure DNS changes succeed.
4. On a domain controller, reset the domain account for the Exchange Virtual Server.
5. Back on the primary cluster node, using Cluster Administrator enable Kerberos on the network name resource and check the "ensure DNS changes succeed box" and then bring the name and disk resources online. Then create an Microsoft Exchange System Attendant resource as needed.
6. 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.
7. Verify DNS points to the correct address on all three branch DNS secondary servers; manually reload zone files and purge DNS cache as needed.
8. Ensure the branch Outlook users begin to receive email again. Check for email until the queues are clear on both sending hosts.
9. Verify no email is lost.

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

- 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-WA passed.

Exchange-NetApp-Fail-Back-Sync-100km-WA

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 datacenter 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-WA test follows:

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 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; 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 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

- 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-WA passed.

Exchange-HP-Fail-Back-Sync-100km-WA

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 datacenter 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-WA test follows:

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 As required, delete the original synchronous CA pairs with a "pairsplit-S" command (with the force option enabled) on the primary data center frame. Then re-establish the SAN replication link(s) as needed and begin resyncing data from the failover data center to the primary data center by creating a new synchronous pair for each device on the failover data center frame.

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 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 Command View GUI or HORCM CLI, delete the pair for each device using a fiber channel path with a "pairsplit-S" command without force enabled.
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 Reestablish replication from the primary to the failover data center.

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

- 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-WA 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.

The following tests were performed:

- Exchange-EMC-Fail-Over-Sync-100km-WA, page 9-28
- Exchange-NetApp-Fail-Over-Sync-100km-WA, page 9-30
- Exchange-HP-Fail-Over-Sync-100km-WA, page 9-31

Exchange-EMC-Fail-Over-Sync-100km-WA

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 datacenter 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-WA test follows:

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: 0. Using Exchange System manager, offline the Public Folder and Mailbox stores.

1. Using Cluster Administrator, offline the Exchange Virtual Server network name resource in the primary Exchange cluster and delete the Microsoft Exchange System Attendant resource (do *not* remove the Exchange Virtual Server). Then offline the Exchange service group.
2. On a host with Solutions Enabler, issue a "symrdf failover" command for the appropriate RDF group and device list. Rescan disks as needed using Disk Manager on the primary fail over cluster node, dcap-xchng-b1.
3. On the primary fail over cluster node, online the Exchange service group in Cluster Administrator, then change the network name resource to *not* use Kerberos or ensure DNS changes succeed.
4. On a domain controller, reset the domain account for the Exchange Virtual Server.
5. Back on the primary fail over cluster node, enable Kerberos on the network name resource and check the "ensure DNS changes succeed box".
6. Using Cluster Administrator on the primary fail over cluster node, create a Microsoft Exchange System Attendant resource using the network name. Use Cluster Administrator to online the Exchange service group. 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).
7. Verify DNS points to the correct address on the DCB master DNS server (fix manually as needed) and all three branch DNS secondary servers; manually reload zone files and purge DNS cache as needed.
8. Ensure the branch Outlook users begin to receive email again. Stop the email scripts, and check for email until the queues are clear on both sending hosts. Flush the email queues on the hosts sending the emails as needed.
9. 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

- 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-WA passed.

Exchange-NetApp-Fail-Over-Sync-100km-WA

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 datacenter 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-WA test follows:

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 and optionally delete the Microsoft Exchange System Attendant resource (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 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.
5. On a domain controller, reset the domain account for the Exchange Virtual Server.
6. Back on the primary fail over cluster node, enable Kerberos on the network name resource and check the "ensure DNS changes succeed box". Then online the Exchange service group and create a Microsoft Exchange System Attendant resource.
7. Using Cluster Administrator on the primary fail over cluster node, online the Exchange service group. 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; 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. 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

- 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-WA passed.

Exchange-HP-Fail-Over-Sync-100km-WA

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 Exchange-HP-Fail-Over-Sync-100km-WA test follows:

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 and optionally delete the Microsoft Exchange System Attendant resource (do *not* remove the Exchange Virtual Server). Then offline the Exchange service group.
3. Using the Command View GUI or HORCM CLI on the primary frame, delete the synchronous CA pairs with a "pairsplit-S" 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 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.
5. On a domain controller, reset the domain account for the Exchange Virtual Server.
6. Back on the primary fail over cluster node, enable Kerberos on the network name resource and check the "ensure DNS changes succeed box". Then online the Exchange service group and create a Microsoft Exchange System Attendant resource.

7. Using Cluster Administrator on the primary fail over cluster node, online the Exchange service group. 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; 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. 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

- 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-WA passed.

CHAPTER 10

Disaster Recovery

DCAP 3.0 testing included disaster recovery testing for the Oracle 11i E-Business Suite, Oracle 10gR2 database, and Microsoft Exchange 2003 applications.

The application topology consists of an Oracle E-Business 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 chapters in this document.

Oracle E-Business 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 two RedHat Enterprise Linux 4 update 4 active/passive database clusters, one in each data center. The two hosts in each cluster access the Oracle executable code from a Network Appliance 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 from EMC, HP, and Network Appliance. 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 fiber channel link to ensure an identical copy of the database is available for failover in DCb.

The Application Tier consists of three non-clustered application hosts in each data center. Two application hosts provide web and Oracle Forms access and the third application host runs the Oracle Concurrent Manager (batch processor). All six 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 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 Cisco 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 Content Switching Module (CSM) in each data center provides load balancing between the two application hosts in each data center by means of a virtual IP (VIP). 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's) in each data center together provide automatic load balancing and failover of the CSM VIP for the application hosts to support the Oracle clients at all three branch locations. The GSS's 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 CSM VIP. In other words, if the CSM VIP goes down in DCa, the remaining GSS's 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 two Microsoft Windows 2003 active/passive back end clusters, one in each data center, which host the Microsoft Exchange 2003 application, an Outlook 2003 client in each of the three branches, and a Linux server in each branch which sends SMTP email to all three clients throughout each test. The primary cluster hosts an Exchange Virtual Server called "DCAP-MBOX-1" and the other cluster acts as a disaster recovery standby cluster. The clusters use fiber channel to attach to storage from EMC, HP, and Network Appliance. 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 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 GSS's 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 GSS's 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 GSS's, 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.

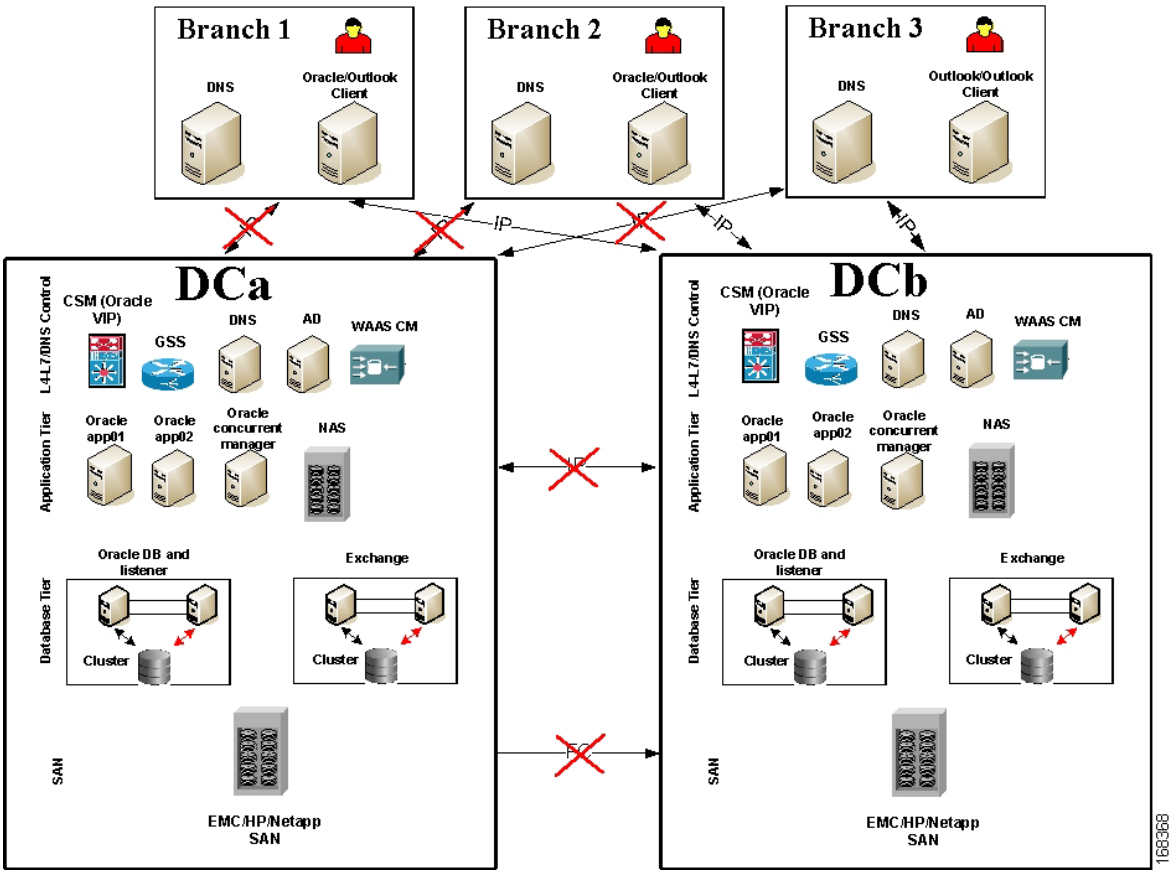
For RPO, at the start of each test, application data is generated by simulated clients (LoadRunner for Oracle and a custom Perl script for Exchange), and a check is made at the completion of failover or failback to ensure all data sent prior to the beginning of the failover or failback is available.

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 10-1 depicts the primary data center components that are important in understanding the DCAP data center disaster recovery failover and failback testing.

Figure 10-1 DCAP Data Center Disaster Recovery Topology



The red X's in the topology diagram show the logical links that are severed when a simulated disaster triggers the failover process.

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 filesystems for the database and application hosts and a shared Oracle data filesystem ("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 Network Appliance, 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 10-1 summarizes the results of failover and failback testing.

Table 10-1 Failover/Failback Test Results

Vendor	Failover		Fallback	
	RPO	RTO	RPO	RTO
EMC	0	44 min	0	97 min
HP	0	148 min	0	30 min
NetApp	0	24 min	0	20 min

The details for each test are shown in a separate timeline.

Figure 10-2 Failover Timeline—EMC

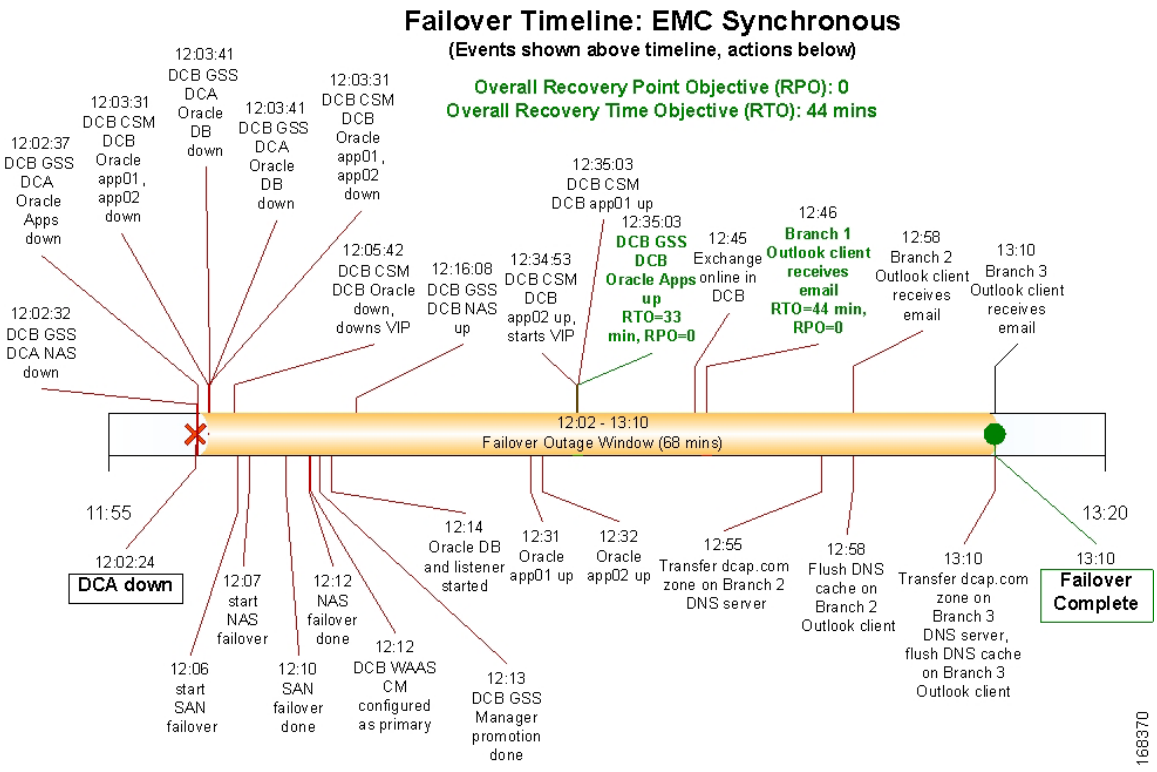


Figure 10-3 Failover Timeline—HP

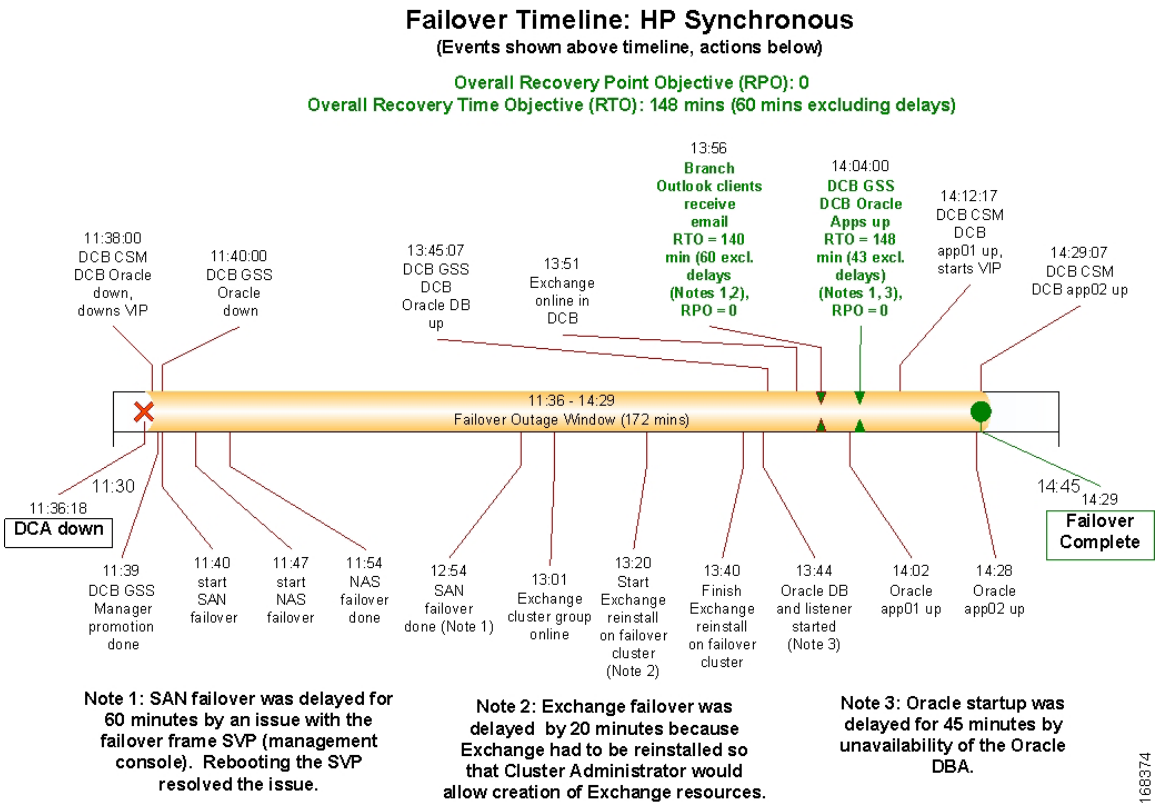


Figure 10-4 Failover Timeline—NetApp

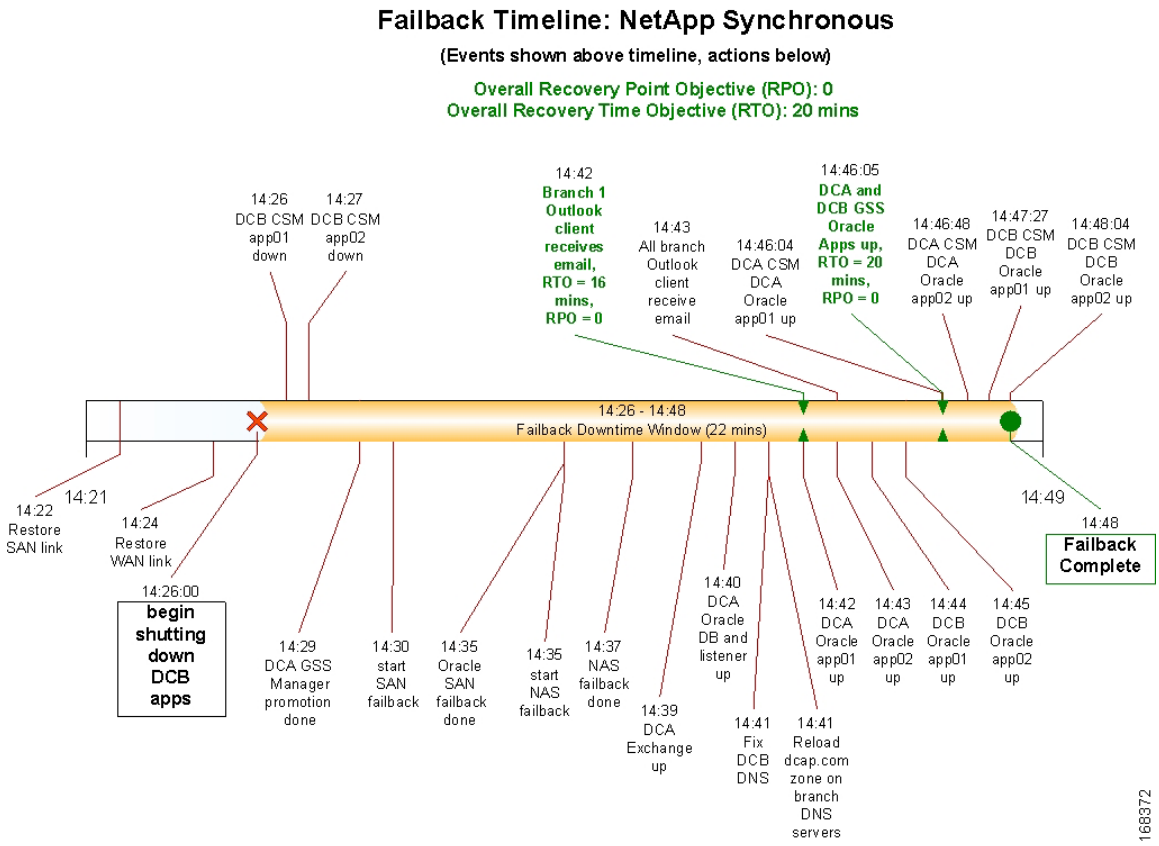


Figure 10-5 Failback Timeline—EMC

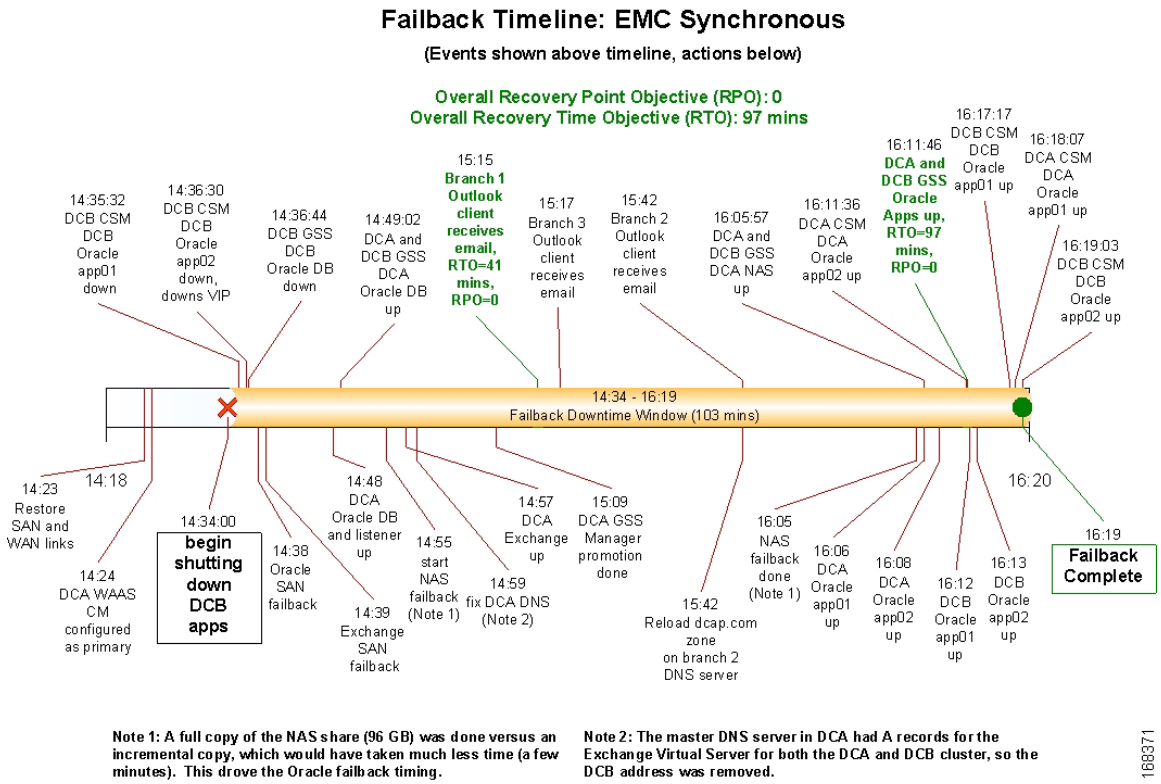


Figure 10-6 Failback Timeline—HP

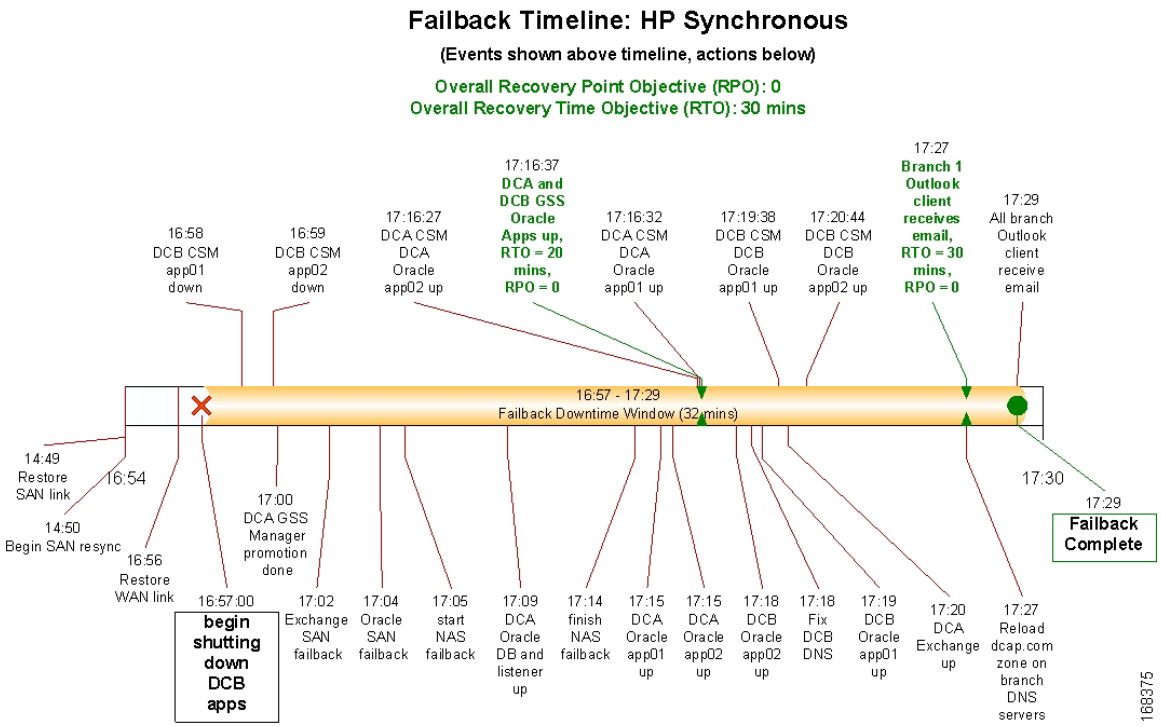
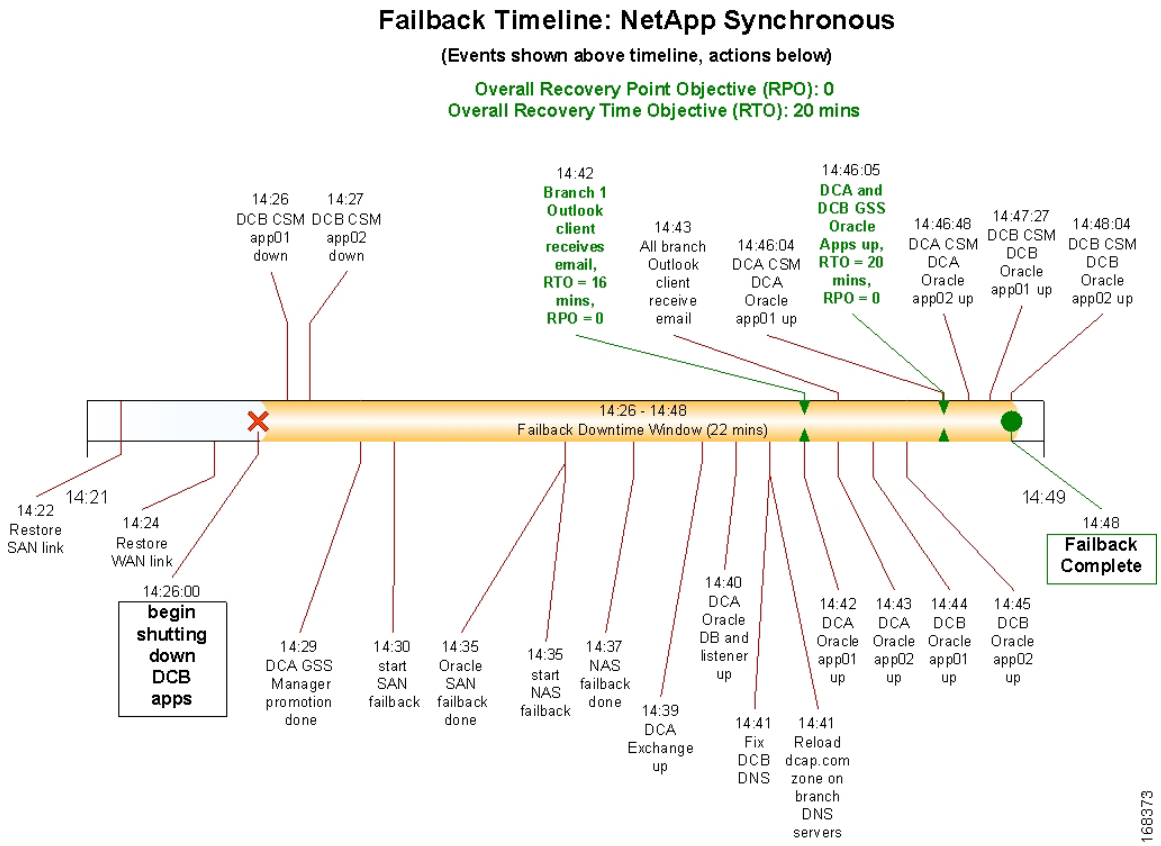


Figure 10-7 Failback Timeline—NetApp



For details about how to perform the failover and failback, please refer to "Disaster Recovery Configuration Details" .

Disaster Recovery Test Results Summary

Table 10-2 summarizes tests executed as part of the Cisco DCAP 3.0 testing initiative. Table 10-2 includes the feature or function tested, the section that describes the feature set the feature or function belongs to, the component tests for each feature or function, and whether the test is new in this phase of DCAP testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested. DCAP is designed to cover critical path areas and augment ongoing regression and systems testing.

Table 10-2 Cisco DCAP 3.0 Disaster Recovery Testing Summary

Test Suites	Features/Functions	Tests	Results
Failover	Failover, page 10-13	1. Disaster Recovery Failover—EMC 2. Disaster Recovery Failover—HP 3. Disaster Recovery Failover—NetApp	
Failback	Failback, page 10-18	1. Disaster Recovery Failback—EMC 2. Disaster Recovery Failback—HP 3. Disaster Recovery Failback—NetApp	

Disaster Recovery Test Cases

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.

- Failover, page 10-13
- Failback, page 10-18

Failover

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.

The following tests were performed:

- Disaster Recovery Failover—EMC, page 10-13
- Disaster Recovery Failover—HP, page 10-15
- Disaster Recovery Failover—NetApp, page 10-16

Disaster Recovery Failover—EMC

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 Tier Layer active in both data centers and Database Tier is active in only one Datacenter). 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 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, 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 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 datacenter failover RTO. Determine the earliest RPO of all applications. This is the datacenter 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

- MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect CSM to automatically load balance Oracle clients to both Oracle application servers in DCB after failover.
- We expect GSS to automatically direct Oracle clients to a sorry server during failover and then direct all clients to DCB after failover.
- We expect all GSS to automatically direct NAS clients to the DCB NAS filer after failover.
- We expect GSS to automatically direct Oracle database clients to the DCB Oracle server after failover.

- We expect all applications to have a Recovery Time Objective (RTO) of less than one hour.
- We expect all applications to have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

Disaster Recovery Failover—EMC passed.

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 Datacenters and Database is Active in only one Datacenter). 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 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 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 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 datacenter failover RTO. Determine the earliest RPO of all applications. This is the datacenter 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

- We expect SAN replication and fail over to be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect CSM to automatically load balance Oracle clients to both Oracle application servers in DCB after failover.
- We expect GSS to automatically direct Oracle clients to a sorry server during failover and then direct all clients to DCB after failover.
- We expect all GSS to automatically direct NAS clients to the DCB NAS filer after failover.
- We expect GSS to automatically direct Oracle database clients to the DCB Oracle server after failover.
- We expect all applications to have a Recovery Time Objective (RTO) of less than one hour.
- We expect all applications to have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

Disaster Recovery Failover—HP passed.

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 Datacenters and Database is Active in only one Datacenter). 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 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 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 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 datacenter failover RTO. Determine the earliest RPO of all applications. This is the datacenter 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

- We expect SAN replication and fail over to be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect CSM to automatically load balance Oracle clients to both Oracle application servers in DCB after failover.
- We expect GSS to automatically direct Oracle clients to a sorry server during failover and then direct all clients to DCB after failover.
- We expect all GSS to automatically direct NAS clients to the DCB NAS filer after failover.
- We expect GSS to automatically direct Oracle database clients to the DCB Oracle server after failover.
- We expect all applications to have a Recovery Time Objective (RTO) of less than one hour.
- We expect all applications to have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).

- We expect no CPU or memory problems.

Results

Disaster Recovery Failover—NetApp passed.

Failback

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.

The following tests were performed:

- Disaster Recovery Failback—EMC, page 10-18
- Disaster Recovery Failback—HP, page 10-20
- Disaster Recovery Failback—NetApp, page 10-21

Disaster Recovery Failback—EMC

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 datacenter. 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 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, 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 datacenter 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 datacenter applications, including the CSM 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 verify all branch clients can receive email (this is the Exchange Recovery Time Objective or RTO). Note the time . 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 Datacenters, 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 datacenter failback RTO. Determine the earliest RPO of all applications. This is the datacenter 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

- We expect SAN replication and failback to be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect all CSM to automatically load balance Oracle clients to both Oracle application servers in both data centers after failback.
- We expect GSS to 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 GSS to automatically direct NAS clients to the DCA NAS filer after failback.
- We expect GSS to automatically direct Oracle database clients to the DCA Oracle server after failback.
- We expect all applications to have a Recovery Time Objective (RTO) of less than one hour.

- We expect all applications to have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

Disaster Recovery Failback—EMC passed.

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 datacenter. 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 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 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 datacenter 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 datacenter applications, including the CSM 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 verify all branch clients can receive email (this is the Exchange Recovery Time Objective or RTO). Note the time . 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 Datacenters, 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 datacenter failback RTO. Determine the earliest RPO of all applications. This is the datacenter 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

- We expect SAN replication and failback to be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect all CSM to automatically load balance Oracle clients to both Oracle application servers in both data centers after failback.
- We expect GSS to 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 GSS to automatically direct NAS clients to the DCA NAS filer after failback.
- We expect GSS to automatically direct Oracle database clients to the DCA Oracle server after failback.
- We expect all applications to have a Recovery Time Objective (RTO) of less than one hour.
- We expect all applications to have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

Disaster Recovery Failback—HP passed.

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 datacenter. 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 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 datacenter 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 datacenter applications, including the CSM 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 verify all branch clients can receive email (this is the Exchange Recovery Time Objective or RTO). Note the time . 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 Datacenters, 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 datacenter failback RTO. Determine the earliest RPO of all applications. This is the datacenter 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

- We expect SAN replication and failback to be supported fully by the MDS replication switches using 100 km of latency and fiber channel write acceleration.
- We expect all CSM to automatically load balance Oracle clients to both Oracle application servers in both data centers after failback.
- We expect GSS to 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 GSS to automatically direct NAS clients to the DCA NAS filer after failback.
- We expect GSS to 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 all applications to have a Recovery Point Objective (RTO) of 0 (due to synchronous SAN and NAS storage replication).
- We expect no CPU or memory problems.

Results

Disaster Recovery Failback—NetApp passed.

APPENDIX A

SAN Configuration Details

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 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 replication.

The load balancing policy for port channels was SID/DID/OXID.

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 device)

Traffic Type	Distance	I/O Type	Host	I/O per sec	MB per sec
Base connectivity	0 km	8 KB sequential writes	Linux	2503	40.0
			Windows	5843	45.7
Synchronous replication	0 km		Linux	1242	9.7
			Windows	1251	9.8
Synchronous replication without FC write acceleration	100 km		Linux	586	4.6
			Windows	585	4.6
Synchronous replication with FC write acceleration	100 km		Linux	908	7.1
			Windows	898	7.0
Asynchronous replication alone	100 km		Linux	4854	37.9
			Windows	5372	41.9
Asynchronous replication with FCIP write acceleration	100 km		Linux	4825	37.7
			Windows	5608	43.8
Asynchronous replication with FCIP compression	100 km		Linux	5058	39.5
			Windows	5890	46.0
Asynchronous replication with FCIP encryption	100 km		Linux	4774	37.3
			Windows	5734	44.8
Asynchronous replication with FCIP write acceleration, compression, and encryption	100 km		Linux	4678	36.6
			Windows	4864	38.0

Table A-2 summarizes the EMC DMX3 software and firmware configuration information.

Table A-2 EMC DMX3 Software/Firmware Information

Software Component	Function	Location	Version
Symmetrix CLI (SYMCLI)	configuration and monitoring, replication control	Linux hosts (x86_64)	V6.3.2.0 (Edit Level: 787)
Symmetrix CLI (SYMCLI)	configuration and monitoring, replication control	Windows hosts (i386)	V6.3.2.0 (Edit Level: 787)
Enginuity Microcode	operating system	frame	5771 (168B0000), Patch date 10.26.2006, Patch level 92
PowerPath	multipath	Linux hosts (x86_64)	5.0.0 (build 157)
PowerPath	multipath	Windows hosts (i386)	4.6.1 (build 5)

Table A-3 summarizes the EMC DMX3 hardware configuration information.

Table A-3 EMC DMX3 Hardware Information

Hardware Component	Quantity	Comments
Frame	2	Serials 000190300320, 000190300321
Cache	16384 MB	per frame
Disk	60 @ 146 GB	per frame
Disk director (DA)	2 @ 2 ports	per frame
Fiber director (FA)	8 @ 2 ports	per frame
Replication director (RA)	4 @ 1 port	per frame

EMC DMX3 Host Device Information

The following EMC DMX3 Host Device Information is available for consideration.

- Windows host dcap-san-hst-05, page A-3
- Linux host dcap-san-hst-06, page A-6
- Windows host dcap-san-hst-07, page A-10
- Linux host dcap-san-hst-08, page A-13

Windows host dcap-san-hst-05

```
DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     D                       DVD-ROM         0 B  Healthy
  Volume 1     C                NTFS   Partition     49 GB  Healthy    System
  Volume 2                              Partition     17 GB  Healthy
  Volume 3                              Partition     17 GB  Healthy
  Volume 4     I   h05-a-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 5     J   h05-a-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 6     G   h05-s-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 7     H   h05-s-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 8     E   h05-l-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 9     F   h05-l-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 10                             Partition     17 GB  Healthy
  Volume 11                             Partition     17 GB  Healthy


C:\Program Files\EMC\PowerPath>powermt display dev=all
Pseudo name=harddisk2
Symmetrix ID=000190300320
Logical device ID=00D3
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
==============================================================================
   6 port6\path0\tgt0\lun41    c6t0d41   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun41    c7t0d41   FA 16cA   active  alive      0      1

Pseudo name=harddisk3
Symmetrix ID=000190300320
```

```
Logical device ID=00D4
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
==============================================================================
   6 port6\path0\tgt0\lun42    c6t0d42   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun42    c7t0d42   FA 16cA   active  alive      0      1

Pseudo name=harddisk4
Symmetrix ID=000190300320
Logical device ID=00D5
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
==============================================================================
   6 port6\path0\tgt0\lun43    c6t0d43   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun43    c7t0d43   FA 16cA   active  alive      0      1

Pseudo name=harddisk5
Symmetrix ID=000190300320
Logical device ID=00D6
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
==============================================================================
   6 port6\path0\tgt0\lun44    c6t0d44   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun44    c7t0d44   FA 16cA   active  alive      0      1

Pseudo name=harddisk6
Symmetrix ID=000190300320
Logical device ID=00E3
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
==============================================================================
   6 port6\path0\tgt0\lun45    c6t0d45   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun45    c7t0d45   FA 16cA   active  alive      0      1

Pseudo name=harddisk7
Symmetrix ID=000190300320
Logical device ID=00E4
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
==============================================================================
   6 port6\path0\tgt0\lun46    c6t0d46   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun46    c7t0d46   FA 16cA   active  alive      0      1

Pseudo name=harddisk8
Symmetrix ID=000190300320
Logical device ID=00DB
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
==============================================================================
   6 port6\path0\tgt0\lun47    c6t0d47   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun47    c7t0d47   FA 16cA   active  alive      0      1
```

```
Pseudo name=harddisk9
Symmetrix ID=000190300320
Logical device ID=00DC
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
==============================================================================
   6 port6\path0\tgt0\lun48    c6t0d48   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun48    c7t0d48   FA 16cA   active  alive      0      1

Pseudo name=harddisk10
Symmetrix ID=000190300320
Logical device ID=00DD
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
==============================================================================
   6 port6\path0\tgt0\lun49    c6t0d49   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun49    c7t0d49   FA 16cA   active  alive      0      1

Pseudo name=harddisk11
Symmetrix ID=000190300320
Logical device ID=00DE
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
==============================================================================
   6 port6\path0\tgt0\lun50    c6t0d50   FA  1cB   active  alive      0     10
   7 port7\path0\tgt0\lun50    c7t0d50   FA 16cA   active  alive      0      1

C:\Program Files\EMC\SYMCLI\bin>symrdf -sid 320 -rdfg 3 -nop -f dcap-san-hst-05_
3_sync.rdf query

Symmetrix ID                       : 000190300320
Remote Symmetrix ID                : 000190300321
RDF (RA) Group Number              :  3 (02)


       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     00D5 RW       0        0 RW 00D5 WD       0        0 S.. Synchronized
N/A     00D6 RW       0        0 RW 00D6 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:\Program Files\EMC\SYMCLI\bin>symrdf -sid 320 -rdfg 4 -nop -f dcap-san-hst-05_
4_async.rdf query

Symmetrix ID                       : 000190300320
Remote Symmetrix ID                : 000190300321
RDF (RA) Group Number              :  4 (03)


       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     00D3 RW       0        0 RW 00D3 WD       0        0 A.. Consistent
N/A     00D4 RW       0        0 RW 00D4 WD       0        0 A.. Consistent

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
```

Linux host dcap-san-hst-06

```
[root@dcap-san-hst-06 ~]# df -k
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/emcpowere1       17399604   4276204  12239536  26% /E
/dev/emcpowerd1       17399604   4276204  12239536  26% /F
/dev/emcpowerj1       17399604   4276204  12239536  26% /G
/dev/emcpoweri1       17399604   4276204  12239536  26% /H
/dev/emcpowerg1       17399604   4276204  12239536  26% /I
/dev/emcpowerh1       17399604   4276204  12239536  26% /J

[root@dcap-san-hst-06 ~]# /sbin/powermt display dev=all
Pseudo name=emcpowerk
Symmetrix ID=000190300320
Logical device ID=008B
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
==============================================================================
   4 qla2xxx                   sdm       FA 16cB   active  alive      0     10

Pseudo name=emcpowerl
Symmetrix ID=000190300320
Logical device ID=008F
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
==============================================================================
   4 qla2xxx                   sdn       FA 16cB   active  alive      0     10
```

```
Pseudo name=emcpowerm
Symmetrix ID=000190300320
Logical device ID=0093
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
==============================================================================
   4 qla2xxx                   sdo       FA 16cB   active  alive      0     10

Pseudo name=emcpowerg
Symmetrix ID=000190300320
Logical device ID=00D7
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
==============================================================================
   4 qla2xxx                   sdc       FA 16cB   active  alive      0     10
   5 qla2xxx                   sds       FA  1cA   active  alive      0      4

Pseudo name=emcpowerh
Symmetrix ID=000190300320
Logical device ID=00D8
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
==============================================================================
   4 qla2xxx                   sdd       FA 16cB   active  alive      0     10
   5 qla2xxx                   sdt       FA  1cA   active  alive      0      4

Pseudo name=emcpowerj
Symmetrix ID=000190300320
Logical device ID=00D9
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
==============================================================================
   4 qla2xxx                   sde       FA 16cB   active  alive      0     10
   5 qla2xxx                   sdu       FA  1cA   active  alive      0      4

Pseudo name=emcpoweri
Symmetrix ID=000190300320
Logical device ID=00DA
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
==============================================================================
   4 qla2xxx                   sdf       FA 16cB   active  alive      0     10
   5 qla2xxx                   sdv       FA  1cA   active  alive      0      4

Pseudo name=emcpowera
Symmetrix ID=000190300320
Logical device ID=00DF
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
==============================================================================
   4 qla2xxx                   sdi       FA 16cB   active  alive      0     10
   5 qla2xxx                   sdy       FA  1cA   active  alive      0      4
```

```
Pseudo name=emcpowerf
Symmetrix ID=000190300320
Logical device ID=00E0
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
==============================================================================
   4 qla2xxx                   sdj       FA 16cB   active  alive      0     10
   5 qla2xxx                   sdz       FA  1cA   active  alive      0      4

Pseudo name=emcpowerc
Symmetrix ID=000190300320
Logical device ID=00E1
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
==============================================================================
   5 qla2xxx                   sdaa      FA  1cA   active  alive      0      4
   4 qla2xxx                   sdk       FA 16cB   active  alive      0     10

Pseudo name=emcpowerb
Symmetrix ID=000190300320
Logical device ID=00E2
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
==============================================================================
   5 qla2xxx                   sdab      FA  1cA   active  alive      0      4
   4 qla2xxx                   sdl       FA 16cB   active  alive      0     10

Pseudo name=emcpowere
Symmetrix ID=000190300320
Logical device ID=00E5
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
==============================================================================
   4 qla2xxx                   sdg       FA 16cB   active  alive      0     10
   5 qla2xxx                   sdw       FA  1cA   active  alive      0      4

Pseudo name=emcpowerd
Symmetrix ID=000190300320
Logical device ID=00E6
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
==============================================================================
   4 qla2xxx                   sdh       FA 16cB   active  alive      0     10
   5 qla2xxx                   sdx       FA  1cA   active  alive      0      4

Pseudo name=emcpowern
Symmetrix ID=000190300320
Logical device ID=00F1
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
==============================================================================
```

```
   4 qla2xxx                   sdp       FA 16cB   active  alive      0     10

Pseudo name=emcpowero
Symmetrix ID=000190300320
Logical device ID=00F4
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
==============================================================================
   4 qla2xxx                   sdq       FA 16cB   active  alive      0     10


[root@dcap-san-hst-06 ~]# /usr/symcli/bin/symrdf -sid 320 -rdfg 3 -nop -f
/devinfo/storage/emc/dcap-san-hst-06_3_sync.rdf query

Symmetrix ID                       : 000190300320
Remote Symmetrix ID                : 000190300321
RDF (RA) Group Number              :  3 (02)



       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     00D9 RW       0        0 RW 00D9 WD       0        0 S.. Synchronized
N/A     00DA RW       0        0 RW 00DA 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


[root@dcap-san-hst-06 ~]# /usr/symcli/bin/symrdf -sid 320 -rdfg 4 -nop -f
/devinfo/storage/emc/dcap-san-hst-06_4_async.rdf query

Symmetrix ID                       : 000190300320
Remote Symmetrix ID                : 000190300321
RDF (RA) Group Number              :  4 (03)



       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     00D7 RW       0        0 RW 00D7 WD       0        0 A.. Consistent
N/A     00D8 RW       0        0 RW 00D8 WD       0        0 A.. Consistent
```

```
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
```

Windows host dcap-san-hst-07

```
DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     D                       DVD-ROM         0 B  Healthy
  Volume 1     C                NTFS   Partition     49 GB  Healthy    System
  Volume 2     G   h07-s-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 3     H   h07-s-emc-5  NTFS   Partition     17 GB  Healthy
  Volume 4                             Partition     17 GB  Healthy
  Volume 5                             Partition     17 GB  Healthy
  Volume 6                             Partition     17 GB  Healthy
  Volume 7                             Partition     17 GB  Healthy
  Volume 8     E   h07-l-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 9     F   h07-l-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 10    I   h07-a-emc-4  NTFS   Partition     17 GB  Healthy
  Volume 11    J   h07-a-emc-4  NTFS   Partition     17 GB  Healthy

C:\Program Files\EMC\PowerPath>powermt display dev=all
Pseudo name=harddisk2
Symmetrix ID=000190300321
Logical device ID=00D3
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
==============================================================================
   6 port6\path0\tgt0\lun41    c6t0d41   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun41    c7t0d41   FA 16cA   active  alive      0      4

Pseudo name=harddisk3
Symmetrix ID=000190300321
Logical device ID=00D4
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
==============================================================================
   6 port6\path0\tgt0\lun42    c6t0d42   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun42    c7t0d42   FA 16cA   active  alive      0      4

Pseudo name=harddisk4
Symmetrix ID=000190300321
Logical device ID=00D5
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
==============================================================================
   6 port6\path0\tgt0\lun43    c6t0d43   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun43    c7t0d43   FA 16cA   active  alive      0      4
```

```
Pseudo name=harddisk5
Symmetrix ID=000190300321
Logical device ID=00D6
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
==============================================================================
   6 port6\path0\tgt0\lun44    c6t0d44   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun44    c7t0d44   FA 16cA   active  alive      0      4

Pseudo name=harddisk6
Symmetrix ID=000190300321
Logical device ID=00E3
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
==============================================================================
   6 port6\path0\tgt0\lun45    c6t0d45   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun45    c7t0d45   FA 16cA   active  alive      0      4

Pseudo name=harddisk7
Symmetrix ID=000190300321
Logical device ID=00E4
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
==============================================================================
   6 port6\path0\tgt0\lun46    c6t0d46   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun46    c7t0d46   FA 16cA   active  alive      0      4

Pseudo name=harddisk8
Symmetrix ID=000190300321
Logical device ID=00DB
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
==============================================================================
   6 port6\path0\tgt0\lun47    c6t0d47   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun47    c7t0d47   FA 16cA   active  alive      0      4

Pseudo name=harddisk9
Symmetrix ID=000190300321
Logical device ID=00DC
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
==============================================================================
   6 port6\path0\tgt0\lun48    c6t0d48   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun48    c7t0d48   FA 16cA   active  alive      0      4

Pseudo name=harddisk10
Symmetrix ID=000190300321
Logical device ID=00DD
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
==============================================================================
```

```
   6 port6\path0\tgt0\lun49    c6t0d49   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun49    c7t0d49   FA 16cA   active  alive      0      4

Pseudo name=harddisk11
Symmetrix ID=000190300321
Logical device ID=00DE
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
==============================================================================
   6 port6\path0\tgt0\lun50    c6t0d50   FA  1cB   active  alive      0      0
   7 port7\path0\tgt0\lun50    c7t0d50   FA 16cA   active  alive      0      4

Pseudo name=harddisk12
Symmetrix ID=000190300321
Logical device ID=0017
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
==============================================================================
   6 port6\path0\tgt0\lun250   c6t0d250  FA  1cB   active  alive      0      0

Pseudo name=harddisk13
Symmetrix ID=000190300321
Logical device ID=001D
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
==============================================================================
   7 port7\path0\tgt0\lun250   c7t0d250  FA 16cA   active  alive      0      4



C:\Program Files\EMC\SYMCLI\bin>symrdf -sid 321 -rdfg 5 -nop -f dcap-san-hst-07
5_sync.rdf query

Symmetrix ID                       : 000190300321
Remote Symmetrix ID                : 000190300320
RDF (RA) Group Number              :  5 (04)



       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     00DD RW       0        0 RW 00DD WD       0        0 S.. Synchronized
N/A     00DE RW       0        0 RW 00DE 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:\Program Files\EMC\SYMCLI\bin>symrdf -sid 321 -rdfg 6 -nop -f dcap-san-hst-07
6_async.rdf query

Symmetrix ID                       : 000190300321
Remote Symmetrix ID                : 000190300320
RDF (RA) Group Number              :  6 (05)



       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     00DB RW       0        0 RW 00DB WD       0        0 A.. Consistent
N/A     00DC RW       0        0 RW 00DC WD       0        0 A.. Consistent

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
```

Linux host dcap-san-hst-08

```
[root@dcap-san-hst-08 ~]# df -k
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/emcpoweri1       17399604   4276204  12239536  26% /E
/dev/emcpowerf1       17399604   4276204  12239536  26% /F
/dev/emcpowerb1       17399604   4276204  12239536  26% /G
/dev/emcpowerc1       17399604   4276204  12239536  26% /H
/dev/emcpowere1       17399604   4276204  12239536  26% /I
/dev/emcpowerh1       17399604   4276204  12239536  26% /J


[root@dcap-san-hst-08 ~]# /sbin/powermt display dev=all
Pseudo name=emcpowerd
Symmetrix ID=000190300321
Logical device ID=0012
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
==============================================================================
   5 lpfc                      sdy       FA  1cA   active  alive      0      1

Pseudo name=emcpowerl
Symmetrix ID=000190300321
Logical device ID=0024
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
==============================================================================
   4 lpfc                      sdm       FA 16cB   active  alive      0      0

Pseudo name=emcpowerk
Symmetrix ID=000190300321
Logical device ID=00D7
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
==============================================================================
   4 lpfc                      sdc       FA 16cB   active  alive      0      0
   5 lpfc                      sdo       FA  1cA   active  alive      0      1

Pseudo name=emcpowerj
Symmetrix ID=000190300321
Logical device ID=00D8
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
==============================================================================
   4 lpfc                      sdd       FA 16cB   active  alive      0      0
   5 lpfc                      sdp       FA  1cA   active  alive      0      1

Pseudo name=emcpowerg
Symmetrix ID=000190300321
Logical device ID=00D9
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
==============================================================================
   4 lpfc                      sde       FA 16cB   active  alive      0      0
   5 lpfc                      sdq       FA  1cA   active  alive      0      1

Pseudo name=emcpowera
Symmetrix ID=000190300321
Logical device ID=00DA
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
==============================================================================
   4 lpfc                      sdf       FA 16cB   active  alive      0      0
   5 lpfc                      sdr       FA  1cA   active  alive      0      1

Pseudo name=emcpowere
Symmetrix ID=000190300321
Logical device ID=00DF
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
==============================================================================
   4 lpfc                      sdi       FA 16cB   active  alive      0      0
   5 lpfc                      sdu       FA  1cA   active  alive      0      1

Pseudo name=emcpowerh
Symmetrix ID=000190300321
Logical device ID=00E0
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
================================================================================
   4 lpfc                      sdj       FA 16cB   active  alive      0      0
   5 lpfc                      sdv       FA  1cA   active  alive      0      1

Pseudo name=emcpowerb
Symmetrix ID=000190300321
Logical device ID=00E1
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
================================================================================
   4 lpfc                      sdk       FA 16cB   active  alive      0      0
   5 lpfc                      sdw       FA  1cA   active  alive      0      1

Pseudo name=emcpowerc
Symmetrix ID=000190300321
Logical device ID=00E2
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
================================================================================
   4 lpfc                      sdl       FA 16cB   active  alive      0      0
   5 lpfc                      sdx       FA  1cA   active  alive      0      1

Pseudo name=emcpoweri
Symmetrix ID=000190300321
Logical device ID=00E5
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
================================================================================
   4 lpfc                      sdg       FA 16cB   active  alive      0      0
   5 lpfc                      sds       FA  1cA   active  alive      0      1

Pseudo name=emcpowerf
Symmetrix ID=000190300321
Logical device ID=00E6
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
================================================================================
   4 lpfc                      sdh       FA 16cB   active  alive      0      0
   5 lpfc                      sdt       FA  1cA   active  alive      0      1


[root@dcap-san-hst-08 ~]# /usr/symcli/bin/symrdf -sid 321 -rdfg 5 -nop -f
/devinfo/storage/emc/dcap-san-hst-08_5_sync.rdf query

Symmetrix ID                       : 000190300321
Remote Symmetrix ID                : 000190300320
RDF (RA) Group Number              :  5 (04)



       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     00E1 RW       0        0 RW 00E1 WD       0        0 S.. Synchronized
N/A     00E2 RW       0        0 RW 00E2 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

[root@dcap-san-hst-08 ~]# /usr/symcli/bin/symrdf -sid 321 -rdfg 6 -nop -f
/devinfo/storage/emc/dcap-san-hst-08_6_async.rdf query

Symmetrix ID                       : 000190300321
Remote Symmetrix ID                : 000190300320
RDF (RA) Group Number              :  6 (05)


       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     00DF RW       0        0 RW 00DF WD       0        0 A.. Consistent
N/A     00E0 RW       0        0 RW 00E0 WD       0        0 A.. Consistent

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
```

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 or replication, but three limitations in NetApp's SAN design are of importance.

1. NetApp's FC-VI-based implementation of SAN-based replication does not allow any benefit from write acceleration over neither fiber channel nor FCIP.

   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. Although 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 3.0 testing showed that it basically works.

   Unfortunately, however, neither FC nor FCIP write acceleration improve performance with NetApp SnapMirror replication. This is because of the FC-VI implementation, which doesn't use the fiber channel protocol 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.

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 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.

   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.

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
Base connectivity	0 km	8 KB sequential writes	Linux	2267	18.1
			Windows	2243	17.5
Synchronous replication	0 km		Linux	1366	10.7
			Windows	1170	9.1
Synchronous replication without FC write acceleration	100 km		Linux	691	5.4
			Windows	698	5.5
Synchronous replication with FC write acceleration	100 km		Linux	677	5.3
			Windows	703	5.5
Asynchronous replication alone	100 km		Linux	2036	15.9
			Windows	2636	20.6
Asynchronous replication with FCIP write acceleration	100 km		Linux	2054	16.0
			Windows	2678	20.9
Asynchronous replication with FCIP compression	100 km		Linux	2028	15.8
			Windows	2631	20.6
Asynchronous replication with FCIP encryption	100 km		Linux	2082	16.3
			Windows	2533	19.8
Asynchronous replication with FCIP write acceleration, compression, and encryption	100 km		Linux	2108	16.5
			Windows	2169	16.9

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

Table A-5 Network Appliance FAS6070 Software/Firmware Information (continued)

Software Component	Function	Location	Version
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	per frame, for base connectivity
FC-VI HBAs	2 @ 2 ports	per frame, for replication; 4 Gbps capable

Network Appliance FAS6070 Device Information

The following Network Appliance FAS6070 device information is available for consideration.

- Windows host dcap-san-hst-01, page A-19
- Linux host dcap-san-hst-02, page A-22
- Windows host dcap-san-hst-03, page A-23
- Linux host dcap-san-hst-04, page A-25

Windows host dcap-san-hst-01

```
DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     D                       DVD-ROM         0 B  Healthy
  Volume 1     G   h01-s-net-2  NTFS   Partition     10 GB  Healthy
  Volume 2     F   h01-l-net-1  NTFS   Partition     10 GB  Healthy
  Volume 3     C                NTFS   Partition    149 GB  Healthy    System
  Volume 4     I   h01-a-net-6  NTFS   Partition     10 GB  Healthy
  Volume 5     E   h01-l-net-0  NTFS   Partition     10 GB  Healthy
  Volume 6     H   h01-s-net-3  NTFS   Partition     10 GB  Healthy
  Volume 7     J   h01-a-net-7  NTFS   Partition     10 GB  Healthy


[DCAP-SAN-HST-01] C:\Program Files\NetApp\MPIO>dsmcli lun attributes
Disks managed by  ONTAPDSM
SerialNumber     Storage System  Storage System Path             MountPath
************     **************  ********************            *********
HnSfSZAcwo53    dcap-netapp-A1   /vol/local1/hst01-lun0 E:\
HnSfSZAcwoNe    dcap-netapp-A1   /vol/local1/hst01-lun1 F:\
```

```
HnSfSZAcwpD8    dcap-netapp-A1   /vol/sync1/hst01-lun2  G:\
HnSfSZAcwpWf    dcap-netapp-A1   /vol/sync1/hst01-lun3  H:\
HnSfSZAcxU/x    dcap-netapp-A1   /vol/async1/hst01-lun6 I:\
HnSfSZAcxUr9    dcap-netapp-A1   /vol/async1/hst01-lun7 J:\


[DCAP-SAN-HST-01] C:\Program Files\NetApp\MPIO>dsmcli path list
Serial Number: HnSfSZAcwo53
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x6000000
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 0
Path State:     ACTIVE

Dsm Id:         0x7000000
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 0
Path State:     PASSIVE

Serial Number: HnSfSZAcwoNe
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x6000001
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 1
Path State:     PASSIVE

Dsm Id:         0x7000001
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 1
Path State:     ACTIVE

Serial Number: HnSfSZAcwpD8
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x6000002
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 2
Path State:     ACTIVE

Dsm Id:         0x7000002
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
```

```
 lun : 2
Path State:     PASSIVE

Serial Number: HnSfSZAcwpWf
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x6000003
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 3
Path State:     PASSIVE

Dsm Id:         0x7000003
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 3
Path State:     ACTIVE

Serial Number: HnSfSZAcxU/x
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x6000006
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 6
Path State:     ACTIVE

Dsm Id:         0x7000006
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 6
Path State:     PASSIVE

Serial Number: HnSfSZAcxUr9
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x6000007
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 7
Path State:     PASSIVE

Dsm Id:         0x7000007
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 7
Path State:     ACTIVE

dcap-netapp-A1> snapmirror status
```

```
Snapmirror is on.
Source                          Destination                   State         Lag
Status
dcap-netapp-A1:async1             dcap-netapp-B1:async1           Source
00:00:37   Idle
dcap-netapp-A1:sync1             dcap-netapp-B1:sync1           Source        -
In-sync
```

Linux host dcap-san-hst-02

```
 [root@dcap-san-hst-02 ~]# /bin/df -k
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/mapper/360a98000486e5365574a41655163756c1
                      10321192   4254240   5542668  44% /E
/dev/mapper/360a98000486e5365574a416551645445p1
                      10321192   4254240   5542668  44% /F
/dev/mapper/360a98000486e5365574a4165516c414c1
                      10321192   4254240   5542668  44% /G
/dev/mapper/360a98000486e5365574a4165516c6430p1
                      10321192   4254240   5542668  44% /H
/dev/mapper/360a98000486e5365574a4165516d4650p1
                      10321192   4254240   5542668  44% /I
/dev/mapper/360a98000486e5365574a4165516d6a43p1
                      10321192   4254240   5542668  44% /J

[root@dcap-san-hst-02 ~]# /sbin/multipath -l
360a98000486e5365574a4165516d6a43
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:5     sdm 8:192 [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:5     sdg 8:96  [active][ready]

360a98000486e5365574a4165516d4650
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:4     sdl 8:176 [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:4     sdf 8:80  [active][ready]

360a98000486e5365574a4165516c414c
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:2     sdj 8:144 [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:2     sdd 8:48  [active][ready]

360a98000486e5365574a41655163756c
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:0     sdh 8:112 [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:0     sdb 8:16  [active][ready]

360a98000486e5365574a4165516c6430
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:3     sdk 8:160 [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:3     sde 8:64  [active][ready]

360a98000486e5365574a416551645445
```

```
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:1     sdi 8:128 [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:1     sdc 8:32  [active][ready]


dcap-netapp-A2> snapmirror status
Snapmirror is on.
Source                                  Destination
State          Lag        Status
dcap-netapp-A2:async2                   dcap-netapp-B2:async2
Source         00:00:11   Idle
```

Windows host dcap-san-hst-03

```
DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     E   h03-l-net-0  NTFS   Partition     10 GB  Healthy
  Volume 1     C                NTFS   Partition    149 GB  Healthy    System
  Volume 2     J   h03-a-net-5  NTFS   Partition     10 GB  Healthy
  Volume 3     I   h03-a-net-4  NTFS   Partition     10 GB  Healthy
  Volume 4     G   h03-s-net-2  NTFS   Partition     10 GB  Healthy
  Volume 5     H   h03-s-net-3  NTFS   Partition     10 GB  Healthy
  Volume 6     D                       DVD-ROM        0 B  Healthy
  Volume 7     F   h03-l-net-1  NTFS   Partition     10 GB  Healthy

[DCAP-SAN-HST-03] C:\Program Files\NetApp\MPIO>dsmcli lun attributes
Disks managed by  ONTAPDSM
SerialNumber     Storage System  Storage System Path            MountPath
************     **************  ********************           *********
HnSehZAgWiKu    dcap-netapp-B1   /vol/local3/hst03-lun0 E:\
HnSehZAgWidn    dcap-netapp-B1   /vol/local3/hst03-lun1 F:\
HnSehZAgWjKZ    dcap-netapp-B1   /vol/sync3/hst03-lun2  G:\
HnSehZAgWj/5    dcap-netapp-B1   /vol/sync3/hst03-lun3  H:\
HnSehZAgWjty    dcap-netapp-B1   /vol/async3/hst03-lun4 I:\
HnSehZAgWkB8    dcap-netapp-B1   /vol/async3/hst03-lun5 J:\

[DCAP-SAN-HST-03] C:\Program Files\NetApp\MPIO>dsmcli path list
Serial Number: HnSehZAgWiKu
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x7000000
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 0
Path State:     ACTIVE

Dsm Id:         0x6000000
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 0
Path State:     PASSIVE

Serial Number: HnSehZAgWidn
```

```
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x7000001
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 1
Path State:     ACTIVE

Dsm Id:         0x6000001
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 1
Path State:     PASSIVE

Serial Number: HnSehZAgWjKZ
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x7000002
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 2
Path State:     ACTIVE

Dsm Id:         0x6000002
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 2
Path State:     PASSIVE

Serial Number: HnSehZAgWj/5
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x7000003
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 3
Path State:     ACTIVE

Dsm Id:         0x6000003
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 3
Path State:     PASSIVE

Serial Number: HnSehZAgWjty
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x7000004
```

```
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 4
Path State:     ACTIVE

Dsm Id:         0x6000004
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 4
Path State:     PASSIVE

Serial Number: HnSehZAgWkB8
MPIO Paths: 2
Load Balance Policy:  FAILOVER

Dsm Id:         0x7000005
SCSI Address:
 Scsiport : 7
 HostPathId : 0
 Targetid : 0
 lun : 5
Path State:     ACTIVE

Dsm Id:         0x6000005
SCSI Address:
 Scsiport : 6
 HostPathId : 0
 Targetid : 0
 lun : 5
Path State:     PASSIVE

dcap-netapp-B1> snapmirror status
Snapmirror is on.
Source                          Destination                     State         Lag
Status
dcap-netapp-B1:async3              dcap-netapp-A1:async3              Source
00:00:48   Idle
```

Linux host dcap-san-hst-04

```
 [root@dcap-san-hst-04 ~]# /bin/df -k
Filesystem           1K-blocks      Used Available Use% Mounted on
                      10321192   4254240   5542668  44% /E
/dev/mapper/360a98000486e53664b34416765435659p1
                      10321192   4254240   5542668  44% /F
/dev/mapper/360a98000486e53664b34416765443453p1
                      10321192   4254240   5542668  44% /G
/dev/mapper/360a98000486e53664b34416765444b32p1
                      10321192   4254240   5542668  44% /H
/dev/mapper/360a98000486e53664b3441676544654a1
                      10321192   4254240   5542668  44% /I
/dev/mapper/360a98000486e53664b34416765453534p1
                      10321192   4254240   5542668  44% /J


[root@dcap-san-hst-04 ~]# /sbin/multipath -l
360a98000486e53664b34416765432d45
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
```

```
\_ round-robin 0 [active]
 \_ 5:0:0:0     sdb 8:16  [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:0     sdh 8:112 [active][ready]

360a98000486e53664b34416765435659
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:1     sdc 8:32  [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:1     sdi 8:128 [active][ready]

360a98000486e53664b34416765443453
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:2     sdd 8:48  [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:2     sdj 8:144 [active][ready]

360a98000486e53664b34416765444b32
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:3     sde 8:64  [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:3     sdk 8:160 [active][ready]

360a98000486e53664b34416765453534
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:5     sdg 8:96  [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:5     sdm 8:192 [active][ready]

360a98000486e53664b3441676544654a
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:4     sdf 8:80  [active][ready]
\_ round-robin 0 [enabled]
 \_ 4:0:0:4     sdl 8:176 [active][ready]


dcap-netapp-B2> snapmirror status
Snapmirror is on.
Source                  Destination             State          Lag        Status
dcap-netapp-B2:async4   dcap-netapp-A2:async4   Source         00:00:41   Idle
dcap-netapp-B2:sync4    dcap-netapp-A2:sync4    Source         -          In-sync
```

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 3.0 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).

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
Base connectivity	0 km	8 KB sequential writes	Linux	6492	46.9
			Windows	5975	46.7
Synchronous replication	0 km		Linux	1364	10.7
			Windows	1341	10.5
Synchronous replication without FC write acceleration	100 km		Linux	560	4.4
			Windows	537	4.2
Synchronous replication with FC write acceleration	100 km		Linux	862	6.7
			Windows	811	6.3
Asynchronous replication alone	100 km		Linux	2070	16.2
			Windows	2284	17.8
Asynchronous replication with FCIP write acceleration	100 km		Linux	2062	16.1
			Windows	2276	17.8
Asynchronous replication with FCIP compression	100 km		Linux	2872	16.2
			Windows	2170	17.0
Asynchronous replication with FCIP encryption	100 km		Linux	2108	16.5
			Windows	2220	17.3
Asynchronous replication with FCIP write acceleration, compression, and encryption	100 km		Linux	2040	15.9
			Windows	2282	17.8

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)
Veritas DMP	multipath	Windows hosts	4.3 (for non-Exchange hosts only)

Table A-8 HP XP10000 Software/Firmware Information (continued)

Software Component	Function	Location	Version
HP MPIO DSM Manager	HP MPIO multipath management	Windows hosts	v2.00.00 (for Exchange hosts only)
HP MPIO Full Featured DSM for XP Disk Arrays	multipath	Windows hosts	v2.00.01 (for Exchange hosts only)

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	per frame
Fiber replication ports	4 ports	per frame
Fiber unused ports	4 ports	per frame

HP XP10000 Device Information

The following HP XP10000 device information is available for consideration.

- Windows host dcap-san-hst-09, page A-29
- Linux host dcap-san-hst-10, page A-36
- Windows host dcap-san-hst-11, page A-38
- Linux host dcap-san-hst-12, page A-45

Windows host dcap-san-hst-09

```
DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     D                       DVD-ROM         0 B  Healthy
  Volume 1     C                NTFS   Partition     49 GB  Healthy    System
  Volume 2     E   h09-l-hp-0   NTFS   Partition     10 GB  Healthy
  Volume 3                              Partition     10 GB  Healthy
  Volume 4                              Partition     10 GB  Healthy
  Volume 5                              Partition     10 GB  Healthy
  Volume 6                              Partition     10 GB  Healthy
  Volume 7     F   h09-l-hp-1   NTFS   Partition     10 GB  Healthy
  Volume 8     G   h09-s-hp-2   NTFS   Partition     10 GB  Healthy
  Volume 9     H   h09-s-hp-3   NTFS   Partition     10 GB  Healthy
  Volume 10    I   h09-a-hp-4   NTFS   Partition     10 GB  Healthy
  Volume 11    J   h09-a-hp-5   NTFS   Partition     10 GB  Healthy
  Volume 12    K   h09-aj-hp-6  NTFS   Partition     10 GB  Healthy
  Volume 13    M   h09-aj-hp-7  NTFS   Partition     10 GB  Healthy
```

```
C:\>vxdmpadm pathinfo harddisk1
==========================================================================
Device information
    Name        :   Harddisk1
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   0
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   0

C:\>vxdmpadm pathinfo harddisk2
==========================================================================
Device information
    Name        :   Harddisk2
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   1
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   1


C:\>vxdmpadm pathinfo harddisk3
==========================================================================
Device information
    Name        :   Harddisk3
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
```

```
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   2
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   2

C:\>vxdmpadm pathinfo harddisk4
==========================================================================
Device information
    Name        :   Harddisk4
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   3
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   3

C:\>vxdmpadm pathinfo harddisk5
==========================================================================
Device information
    Name        :   Harddisk5
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   4
Path 7-0-0:
```

```
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   4

C:\>vxdmpadm pathinfo harddisk6
==========================================================================
Device information
    Name        :   Harddisk6
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   5
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   5

C:\>vxdmpadm pathinfo harddisk7
==========================================================================
Device information
    Name        :   Harddisk7
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   6
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   6

C:\>vxdmpadm pathinfo harddisk8
```

```
==========================================================================
Device information
    Name        :   Harddisk8
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   7
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   7

C:\>vxdmpadm pathinfo harddisk9
==========================================================================
Device information
    Name        :   Harddisk9
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   18
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   18

C:\>vxdmpadm pathinfo harddisk10
==========================================================================
Device information
    Name        :   Harddisk10
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
```

```
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   19
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   19

C:\>vxdmpadm pathinfo harddisk11
=========================================================================
Device information
    Name        :   Harddisk11
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   20
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   20

C:\>vxdmpadm pathinfo harddisk12
=========================================================================
Device information
    Name        :   Harddisk12
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   21
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
```

```
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   21

C:\>vxdmpadm pathinfo harddisk13
==========================================================================
Device information
    Name        :   Harddisk13
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   22
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   22

[root@dcap-san-hst-10 ~]# /usr/bin/pairdisplay -I0 -g dcap-san-hst-09-s -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-09-s      sync-09-1(L) (CL1-A , 0,   1)82836    11.P-VOL PAIR NEVER ,82931
11 -
dcap-san-hst-09-s      sync-09-1(R) (CL1-A , 0,   1)82836    11.P-VOL PAIR NEVER ,82931
11 -
dcap-san-hst-09-s      sync-09-2(L) (CL1-A , 0,   3)82836    17.P-VOL PAIR NEVER ,82931
17 -
dcap-san-hst-09-s      sync-09-2(R) (CL1-A , 0,   3)82836    17.P-VOL PAIR NEVER ,82931
17 -
[root@dcap-san-hst-10 ~]# /usr/bin/pairdisplay -I0 -g dcap-san-hst-09-a -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-09-a      async-09-1(L) (CL1-A , 0,   5)82836   110.P-VOL PAIR ASYNC ,82931
110 -
dcap-san-hst-09-a      async-09-1(R) (CL1-A , 0,   5)82836   110.P-VOL PAIR ASYNC ,82931
110 -
dcap-san-hst-09-a      async-09-2(L) (CL1-A , 0,   7)82836   116.P-VOL PAIR ASYNC ,82931
116 -
dcap-san-hst-09-a      async-09-2(R) (CL1-A , 0,   7)82836   116.P-VOL PAIR ASYNC ,82931
116 -
[root@dcap-san-hst-10 ~]# /usr/bin/pairdisplay -I0 -g dcap-san-hst-09-j -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-09-j      asyncj-09-1(L) (CL1-A , 0,   6)82836   111.SMPL  ---- ------,-----
----- -
dcap-san-hst-09-j      asyncj-09-1(R) (CL1-A , 0,   6)82836   111.SMPL  ---- ------,-----
----- -
dcap-san-hst-09-j      asyncj-09-2(L) (CL1-A , 0,   8)82836   117.SMPL  ---- ------,-----
----- -
dcap-san-hst-09-j      asyncj-09-2(R) (CL1-A , 0,   8)82836   117.SMPL  ---- ------,-----
----- -
```

Linux host dcap-san-hst-10

```
 [root@dcap-san-hst-10 ~]# /bin/df -k
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/mapper/360060e8014439400000143940000001c1
                      10516712   4257316   5725176  43% /E
/dev/mapper/360060e8014439400000143940000022p1
                      10516712   4257316   5725176  43% /F
/dev/mapper/360060e8014439400000143940000001d1
                      10516712   4257316   5725176  43% /G
/dev/mapper/360060e8014439400000143940000023p1
                      10516712   4257316   5725176  43% /H
/dev/mapper/360060e8014439400000143940000011c1
                      10516712   4257316   5725176  43% /I
/dev/mapper/360060e8014439400000143940000122p1
                      10516712   4257316   5725176  43% /J
/dev/mapper/360060e8014439400000143940000011d1
                      10516712   4257316   5725176  43% /K
/dev/mapper/360060e8014439400000143940000123p1
                      10516712   4257316   5725176  43% /M

[root@dcap-san-hst-10 ~]# /sbin/multipath -l
360060e8014439400000143940000063
[size=46 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:242   sdaj 66:48  [active][ready]
 \_ 4:0:0:242   sdr  65:16  [active][ready]

360060e8014439400000143940000163
[size=46 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:243   sdak 66:64  [active][ready]
 \_ 4:0:0:243   sds  65:32  [active][ready]

360060e8014439400000143940000011f
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:22    sdaf 65:240 [active][ready]
 \_ 4:0:0:22    sdn  8:208  [active][ready]

360060e8014439400000143940000061
[size=46 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:240   sdah 66:16  [active][ready]
 \_ 4:0:0:240   sdp  8:240  [active][ready]

360060e8014439400000143940000011e
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:20    sdad 65:208 [active][ready]
 \_ 4:0:0:20    sdl  8:176  [active][ready]

360060e8014439400000143940000161
[size=46 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:241   sdai 66:32  [active][ready]
 \_ 4:0:0:241   sdq  65:0   [active][ready]

360060e8014439400000143940000001d
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 4:0:0:2     sdd  8:48   [active][ready]
 \_ 5:0:0:2     sdv  65:80  [active][ready]
```

```
360060e8014439400000143940000011d
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 4:0:0:6     sdh  8:112  [active][ready]
 \_ 5:0:0:6     sdz  65:144 [active][ready]

360060e8014439400000143940000025
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:19    sdac 65:192 [active][ready]
 \_ 4:0:0:19    sdk  8:160  [active][ready]

360060e8014439400000143940000001c
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 4:0:0:0     sdb  8:16   [active][ready]
 \_ 5:0:0:0     sdt  65:48  [active][ready]

360060e8014439400000143940000125
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:23    sdag 66:0   [active][ready]
 \_ 4:0:0:23    sdo  8:224  [active][ready]

360060e8014439400000143940000011c
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 4:0:0:4     sdf  8:80   [active][ready]
 \_ 5:0:0:4     sdx  65:112 [active][ready]

360060e8014439400000143940000124
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:21    sdae 65:224 [active][ready]
 \_ 4:0:0:21    sdm  8:192  [active][ready]

360060e8014439400000143940000023
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 4:0:0:3     sde  8:64   [active][ready]
 \_ 5:0:0:3     sdw  65:96  [active][ready]

360060e8014439400000143940000123
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:7     sdaa 65:160 [active][ready]
 \_ 4:0:0:7     sdi  8:128  [active][ready]

360060e8014439400000143940000022
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 4:0:0:1     sdc  8:32   [active][ready]
 \_ 5:0:0:1     sdu  65:64  [active][ready]

360060e8014439400000143940000122
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 4:0:0:5     sdg  8:96   [active][ready]
 \_ 5:0:0:5     sdy  65:128 [active][ready]

[root@dcap-san-hst-10 ~]# /usr/bin/pairdisplay -I0 -g dcap-san-hst-10-s -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
```

```
dcap-san-hst-10-s      sync-10-1(L) (CL2-A , 0,   1)82836    1d.P-VOL PAIR NEVER ,82931
1d -
dcap-san-hst-10-s      sync-10-1(R) (CL2-A , 0,   1)82836    1d.P-VOL PAIR NEVER ,82931
1d -
dcap-san-hst-10-s      sync-10-2(L) (CL2-A , 0,   3)82836    23.P-VOL PAIR NEVER ,82931
23 -
dcap-san-hst-10-s      sync-10-2(R) (CL2-A , 0,   3)82836    23.P-VOL PAIR NEVER ,82931
23 -

[root@dcap-san-hst-10 ~]# /usr/bin/pairdisplay -I0 -g dcap-san-hst-10-a -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-10-a      async10-1(L) (CL2-A , 0,   5)82836   11c.P-VOL PAIR ASYNC ,82931
11c -
dcap-san-hst-10-a      async10-1(R) (CL2-A , 0,   5)82836   11c.P-VOL PAIR ASYNC ,82931
11c -
dcap-san-hst-10-a      async10-2(L) (CL2-A , 0,   7)82836   122.P-VOL PAIR ASYNC ,82931
122 -
dcap-san-hst-10-a      async10-2(R) (CL2-A , 0,   7)82836   122.P-VOL PAIR ASYNC ,82931
122 -

[root@dcap-san-hst-10 ~]# /usr/bin/pairdisplay -I0 -g dcap-san-hst-10-j -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-10-j      asyncj-10-1(L) (CL2-A , 0,   6)82836   11d.SMPL  ---- ------,-----
----- -
dcap-san-hst-10-j      asyncj-10-1(R) (CL2-A , 0,   6)82836   11d.SMPL  ---- ------,-----
----- -
dcap-san-hst-10-j      asyncj-10-2(L) (CL2-A , 0,   8)82836   123.SMPL  ---- ------,-----
----- -
dcap-san-hst-10-j      asyncj-10-2(R) (CL2-A , 0,   8)82836   123.SMPL  ---- ------,-----
----- -
```

Windows host dcap-san-hst-11

```
DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     D                       DVD-ROM         0 B  Healthy
  Volume 1     C                NTFS   Partition     49 GB  Healthy    System
  Volume 2     E   h11-l-hp-0   NTFS   Partition     10 GB  Healthy
  Volume 3                             Partition     10 GB  Healthy
  Volume 4                             Partition     10 GB  Healthy
  Volume 5                             Partition     10 GB  Healthy
  Volume 6                             Partition     10 GB  Healthy
  Volume 7     F   h11-l-hp-1   NTFS   Partition     10 GB  Healthy
  Volume 8     G   h11-s-hp-2   NTFS   Partition     10 GB  Healthy
  Volume 9     H   h11-s-hp-3   NTFS   Partition     10 GB  Healthy
  Volume 10    I   h11-a-hp-4   NTFS   Partition     10 GB  Healthy
  Volume 11    J   h11-a-hp-5   NTFS   Partition     10 GB  Healthy
  Volume 12    K   h11-aj-hp-6  NTFS   Partition     10 GB  Healthy
  Volume 13    M   h11-aj-hp-7  NTFS   Partition     10 GB  Healthy

C:\>vxdisk list
Name             MediaName   Diskgroup         DiskStyle  Size(MB)  FreeSpace
)   Status
Harddisk0                   BasicGroup           MBR      152625     102626
Uninitialized
Harddisk1                   BasicGroup           MBR      10432      0
Uninitialized
Harddisk10                  BasicGroup           MBR      10432      0
Uninitialized
Harddisk11                  BasicGroup           MBR      10432      0
```

```
Uninitialized
Harddisk12                  BasicGroup           MBR      10432      0
Uninitialized
Harddisk13                  BasicGroup           MBR      10432      0
Uninitialized
Harddisk14                  BasicGroup           MBR      10432      0
Uninitialized
Harddisk2                   BasicGroup           MBR      10432      0
Uninitialized
Harddisk3                   BasicGroup           MBR      10432      0
Uninitialized
Harddisk4                   BasicGroup           MBR      10432      0
Uninitialized
Harddisk5                   BasicGroup           MBR      10432      0
Uninitialized
Harddisk6                   BasicGroup           MBR      10432      0
Uninitialized
Harddisk7                   BasicGroup           MBR      10432      0
Uninitialized
Harddisk8                   BasicGroup           MBR      10432      0
Uninitialized
Harddisk9                   BasicGroup           MBR      10432      0
Uninitialized


C:\>vxdmpadm pathinfo harddisk1
==========================================================================
Device information
    Name        :   Harddisk1
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   0
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   0

C:\>vxdmpadm pathinfo harddisk2
==========================================================================
Device information
    Name        :   Harddisk2
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
```

```
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   1
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   1

C:\>vxdmpadm pathinfo harddisk3
=========================================================================
Device information
    Name        :   Harddisk3
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   2
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   2

C:\>vxdmpadm pathinfo harddisk4
=========================================================================
Device information
    Name        :   Harddisk4
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   3
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
```

```
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   3

C:\>vxdmpadm pathinfo harddisk5
==========================================================================
Device information
    Name        :   Harddisk5
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   4
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   4

C:\>vxdmpadm pathinfo harddisk6
==========================================================================
Device information
    Name        :   Harddisk6
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   5
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   5

C:\>vxdmpadm pathinfo harddisk7
==========================================================================
Device information
    Name        :   Harddisk7
    Media Name  :
```

```
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   6
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   6

C:\>vxdmpadm pathinfo harddisk8
==========================================================================
Device information
    Name        :   Harddisk8
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   7
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   7

C:\>vxdmpadm pathinfo harddisk9
==========================================================================
Device information
    Name        :   Harddisk9
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
```

```
    Target      :   0
    LUN         :   18
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   18

C:\>vxdmpadm pathinfo harddisk10
==========================================================================
Device information
    Name        :   Harddisk10
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   19
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   19

C:\>vxdmpadm pathinfo harddisk11
==========================================================================
Device information
    Name        :   Harddisk11
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   20
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
```

```
    LUN         :   20

C:\>vxdmpadm pathinfo harddisk12
==========================================================================
Device information
    Name        :   Harddisk12
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   21
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   21

C:\>vxdmpadm pathinfo harddisk13
==========================================================================
Device information
    Name        :   Harddisk13
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   22
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   22

C:\>vxdmpadm pathinfo harddisk14
==========================================================================
Device information
    Name        :   Harddisk14
    Media Name  :
    No. of Paths:   2
    LoadBalancePolicy       :   Round Robin (Active/Active)
Path information ...
```

```
Path 6-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   6
    Channel     :   0
    Target      :   0
    LUN         :   23
Path 7-0-0:
    State       :   Healthy
    Primary     :   NO
    SCSI-3 Persistent Reservation: YES
    SCSI-3 Reserved: NO
    Port        :   7
    Channel     :   0
    Target      :   0
    LUN         :   23

[root@dcap-san-hst-12 ~]# /usr/bin/pairdisplay -I0  -g dcap-san-hst-11-s -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-11-s      sync-11-1(L) (CL1-A , 0,   1)82931    13.P-VOL PAIR NEVER ,82836
13 -
dcap-san-hst-11-s      sync-11-1(R) (CL1-A , 0,   1)82931    13.P-VOL PAIR NEVER ,82836
13 -
dcap-san-hst-11-s      sync-11-2(L) (CL1-A , 0,   3)82931    19.P-VOL PAIR NEVER ,82836
19 -
dcap-san-hst-11-s      sync-11-2(R) (CL1-A , 0,   3)82931    19.P-VOL PAIR NEVER ,82836
19 -
[root@dcap-san-hst-12 ~]# /usr/bin/pairdisplay -I0  -g dcap-san-hst-11-a -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-11-a      async-11-1(L) (CL1-A , 0,   5)82931   112.P-VOL PAIR ASYNC ,82836
112 -
dcap-san-hst-11-a      async-11-1(R) (CL1-A , 0,   5)82931   112.P-VOL PAIR ASYNC ,82836
112 -
dcap-san-hst-11-a      async-11-2(L) (CL1-A , 0,   7)82931   118.P-VOL PAIR ASYNC ,82836
118 -
dcap-san-hst-11-a      async-11-2(R) (CL1-A , 0,   7)82931   118.P-VOL PAIR ASYNC ,82836
118 -
[root@dcap-san-hst-12 ~]# /usr/bin/pairdisplay -I0  -g dcap-san-hst-11-j -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-11-j       asyncj-11-1(L) (CL1-A , 0,   6)82931   113.SMPL  ---- ------,-----
----- -
dcap-san-hst-11-j       asyncj-11-1(R) (CL1-A , 0,   6)82931   113.SMPL  ---- ------,-----
----- -
dcap-san-hst-11-j       asyncj-11-2(L) (CL1-A , 0,   8)82931   119.SMPL  ---- ------,-----
----- -
dcap-san-hst-11-j       asyncj-11-2(R) (CL1-A , 0,   8)82931   119.SMPL  ---- ------,-----
----- -
```

Linux host dcap-san-hst-12

```
[root@dcap-san-hst-12 ~]# /bin/df -k
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/mapper/360060e801443f300000143f30000001e1
                      10516712   4257316   5725176  43% /E
/dev/mapper/360060e801443f300000143f300000024p1
                      10516712   4257316   5725176  43% /F
/dev/mapper/360060e801443f300000143f30000001f1
                      10516712   4257316   5725176  43% /G
/dev/mapper/360060e801443f300000143f300000025p1
                      10516712   4257316   5725176  43% /H
```

```
/dev/mapper/360060e801443f300000143f30000011e1
                      10516712   4257316   5725176  43% /I
/dev/mapper/360060e801443f300000143f300000124p1
                      10516712   4257316   5725176  43% /J
/dev/mapper/360060e801443f300000143f30000011f1
                      10516712   4257316   5725176  43% /K
/dev/mapper/360060e801443f300000143f300000125p1
                      10516712   4257316   5725176  43% /M

[root@dcap-san-hst-12 ~]# /sbin/multipath -l
360060e801443f300000143f300000163
[size=46 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:241   sdai 66:32  [active][ready]
 \_ 4:0:0:241   sdq  65:0   [active][ready]

360060e801443f300000143f30000001f
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 4:0:0:2     sdd  8:48   [active][ready]
 \_ 5:0:0:2     sdv  65:80  [active][ready]

360060e801443f300000143f30000011f
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:6     sdz  65:144 [active][ready]
 \_ 4:0:0:6     sdh  8:112  [active][ready]

360060e801443f300000143f30000001e
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:0     sdt  65:48  [active][ready]
 \_ 4:0:0:0     sdb  8:16   [active][ready]

360060e801443f300000143f300000061
[size=46 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:242   sdaj 66:48  [active][ready]
 \_ 4:0:0:242   sdr  65:16  [active][ready]

360060e801443f300000143f30000011e
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:4     sdx  65:112 [active][ready]
 \_ 4:0:0:4     sdf  8:80   [active][ready]

360060e801443f300000143f30000001d
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:18    sdab 65:176 [active][ready]
 \_ 4:0:0:18    sdj  8:144  [active][ready]

360060e801443f300000143f30000011d
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:22    sdaf 65:240 [active][ready]
 \_ 4:0:0:22    sdn  8:208  [active][ready]

360060e801443f300000143f300000025
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:3     sdw  65:96  [active][ready]
 \_ 4:0:0:3     sde  8:64   [active][ready]
```

```
360060e801443f300000143f300000125
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:7     sdaa 65:160 [active][ready]
 \_ 4:0:0:7     sdi  8:128  [active][ready]

360060e801443f300000143f30000011c
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:20    sdad 65:208 [active][ready]
 \_ 4:0:0:20    sdl  8:176  [active][ready]

360060e801443f300000143f300000024
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:1     sdu  65:64  [active][ready]
 \_ 4:0:0:1     sdc  8:32   [active][ready]

360060e801443f300000143f300000124
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:5     sdy  65:128 [active][ready]
 \_ 4:0:0:5     sdg  8:96   [active][ready]

360060e801443f300000143f300000023
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:19    sdac 65:192 [active][ready]
 \_ 4:0:0:19    sdk  8:160  [active][ready]

360060e801443f300000143f300000123
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:23    sdag 66:0   [active][ready]
 \_ 4:0:0:23    sdo  8:224  [active][ready]

360060e801443f300000143f300000122
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 5:0:0:21    sdae 65:224 [active][ready]
 \_ 4:0:0:21    sdm  8:192  [active][ready]


[root@dcap-san-hst-12 ~]# /usr/bin/pairdisplay -I0  -g dcap-san-hst-12-s -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-12-s      sync-12-1(L) (CL2-A , 0,   1)82931    1f.P-VOL PAIR NEVER ,82836
1f -
dcap-san-hst-12-s      sync-12-1(R) (CL2-A , 0,   1)82931    1f.P-VOL PAIR NEVER ,82836
1f -
dcap-san-hst-12-s      sync-12-2(L) (CL2-A , 0,   3)82931    25.P-VOL PAIR NEVER ,82836
25 -
dcap-san-hst-12-s      sync-12-2(R) (CL2-A , 0,   3)82931    25.P-VOL PAIR NEVER ,82836
25 -
[root@dcap-san-hst-12 ~]# /usr/bin/pairdisplay -I0  -g dcap-san-hst-12-a -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-12-a      async-12-1(L) (CL2-A , 0,   5)82931   11e.P-VOL PAIR ASYNC ,82836
11e -
dcap-san-hst-12-a      async-12-1(R) (CL2-A , 0,   5)82931   11e.P-VOL PAIR ASYNC ,82836
11e -
dcap-san-hst-12-a      async-12-2(L) (CL2-A , 0,   7)82931   124.P-VOL PAIR ASYNC ,82836
124 -
dcap-san-hst-12-a      async-12-2(R) (CL2-A , 0,   7)82931   124.P-VOL PAIR ASYNC ,82836
124 -
[root@dcap-san-hst-12 ~]# /usr/bin/pairdisplay -I0  -g dcap-san-hst-12-j -fx
```

```
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-san-hst-12-j       asyncj-12-1(L) (CL2-A , 0,   6)82931   11f.SMPL  ---- ------,-----
----- -
dcap-san-hst-12-j       asyncj-12-1(R) (CL2-A , 0,   6)82931   11f.SMPL  ---- ------,-----
----- -
dcap-san-hst-12-j       asyncj-12-2(L) (CL2-A , 0,   8)82931   125.SMPL  ---- ------,-----
----- -
dcap-san-hst-12-j       asyncj-12-2(R) (CL2-A , 0,   8)82931   125.SMPL  ---- ------,-----
----- -
```

ADIC

This section has general and detailed information about the 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 ADIC tape library, but a few apparent anomalies need to be explained.

Local Baseline Slower Than Remote Baseline

In the local baseline tests, a backup (write) or restore (read) of the test filesystem to a tape drive connected via fiber channel with negligible latency (that is, locally) was somewhat slower than a backup or restore done over fiber channel with negligible latency. Even though this might be counterintuitive and there's no clear explanation for it at this time, the remote baseline test can safely be considered as the fastest possible throughput without FCIP write acceleration enabled.

Compression Did Not Improve Throughput

Except for the read acceleration tests at 0 and 100 km and the write acceleration test at 5000 km, neither software nor hardware compression significantly improved throughput. This can be explained by the fact that the FCIP tunnels each had 311 Mbps of throughput (half an OC-12) availabled to them, so bandwidth was not constrained. In a bandwidth-constrained environment, compression should improve throughput. Future DCAP testing will incorporate lower bandwidth tests.

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.

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 ADIC Scalar i500 Backup/Restore Results

Traffic Type	Distance	I/O type	Host	Time (MM:SS)	MB per sec
Local Baseline	0 km	read	Linux	11:29	10.82
Remote Baseline	0 km	read	Linux	10:31	11.83
FCIP tape acceleration enabled, no compression	0 km	read	Linux	10:25	11.94
FCIP tape acceleration enabled, no compression	100 km	read	Linux	10:55	11.40
FCIP tape acceleration enabled, no compression	5000 km	read	Linux	12:25	10.02
FCIP tape acceleration enabled, hardware compression	0 km	read	Linux	9:29	13.12
FCIP tape acceleration enabled, hardware compression	100 km	read	Linux	9:25	13.21
FCIP tape acceleration enabled, hardware compression	5000 km	read	Linux	13:19	9:34
FCIP tape acceleration enabled, software compression	0 km	read	Linux	25:01	4.97
FCIP tape acceleration enabled, software compression	100 km	read	Linux	25.01	4.97
FCIP tape acceleration enabled, software compression	5000 km	read	Linux	24.53	5.00
Local Baseline	0 km	write	Linux	8:01	14.45
Remote Baseline	0 km	write	Linux	7:20	15.82
FCIP tape acceleration enabled, no compression	0 km	write	Linux	8:35	13.52
FCIP tape acceleration enabled, no compression	100 km	write	Linux	8:12	14.15
FCIP tape acceleration enabled, no compression	5000 km	write	Linux	10:00	11.60
FCIP tape acceleration enabled, hardware compression	0 km	write	Linux	8:24	13.81
FCIP tape acceleration enabled, hardware compression	100 km	write	Linux	8:05	14.35
FCIP tape acceleration enabled, hardware compression	5000 km	write	Linux	8:11	14:18
FCIP tape acceleration enabled, software compression	0 km	write	Linux	24:28	4.74
FCIP tape acceleration enabled, software compression	100 km	write	Linux	25:06	4.62
FCIP tape acceleration enabled, software compression	5000 km	write	Linux	22:30	5.16

Table A-11 summarizes the ADIC Scalar i500 software and firmware configuration information.

Table A-11 ADIC 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

Table A-12 summarizes the HP XP10000 hardware configuration information.

Table A-12 ADIC 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)

ADIC Scalar i500 Host Information

The following ADIC Scalar i500 host information is available for consideration.

- Linux host dcap-dca-oradb02 (local to tape library in DCa), page A-50
- Linux host dcap-dcb-oradb02 (remote in DCb), page A-51

Linux host dcap-dca-oradb02 (local to tape library in DCa)

```
[root@dcap-dca-oradb02-m bin]# pwd
/usr/openv/netbackup/bin
[root@dcap-dca-oradb02-m bin]# cat version
NetBackup-RedHat2.4 6.0

[root@dcap-dca-oradb02-m bin]# /usr/openv/volmgr/bin/tpconfig -l
Device Robot Drive       Robot                    Drive                Device     Second
Type     Num Index  Type DrNum Status  Comment    Name                 Path       Device
Path
robot      0    -    TLD    -       -  -          -                    /dev/sg64
  drive    -    0 hcart3    1      UP  -          IBM.ULTRIUM-TD3.000  /dev/nst1
  drive    -    1 hcart3    2      UP  -          IBM.ULTRIUM-TD3.001  /dev/nst0

[root@dcap-dca-oradb02-m bin]# /usr/openv/volmgr/bin/tpconfig -d
Id  DriveName           Type   Residence
      Drive Path                                                       Status
****************************************************************************
0   IBM.ULTRIUM-TD3.000  hcart3 TLD(0)  DRIVE=1
      /dev/nst1                                                        UP
1   IBM.ULTRIUM-TD3.001  hcart3 TLD(0)  DRIVE=2
      /dev/nst0                                                        UP
```

```
Currently defined robotics are:
  TLD(0)     robotic path = /dev/sg64,

EMM Server = dcap-dca-oradb02

[root@dcap-dca-oradb02-m ~]# df -k /tapetest
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/cciss/c0d0p8      8254240   7633836    201112  98% /tapetest

[root@dcap-dca-oradb02-m ~]# df -i /tapetest
Filesystem            Inodes   IUsed   IFree IUse% Mounted on
/dev/cciss/c0d0p8    1048576  217863  830713   21% /tapetest
```

Linux host dcap-dcb-oradb02 (remote in DCb)

```
[root@dcap-dcb-oradb02-m bin]# pwd
/usr/openv/netbackup/bin
[root@dcap-dcb-oradb02-m bin]# cat version
NetBackup-RedHat2.4 6.0

[root@dcap-dcb-oradb02-m bin]# /usr/openv/volmgr/bin/tpconfig -l
Device Robot Drive       Robot                    Drive                Device     Second
Type     Num Index  Type DrNum Status  Comment    Name                 Path       Device
Path
robot      0    -    TLD    -       -  -          -                    dcap-dca-oradb02
  drive    -    0 hcart3    2      UP  -          IBM.ULTRIUM-TD3.001  /dev/nst1
  drive    -    1 hcart3    1      UP  -          IBM.ULTRIUM-TD3.000  /dev/nst0
[root@dcap-dcb-oradb02-m bin]# /usr/openv/volmgr/bin/tpconfig -d
Id  DriveName           Type   Residence
      Drive Path                                                       Status
****************************************************************************
0   IBM.ULTRIUM-TD3.001  hcart3 TLD(0)  DRIVE=2
      /dev/nst1                                                        UP
1   IBM.ULTRIUM-TD3.000  hcart3 TLD(0)  DRIVE=1
      /dev/nst0                                                        UP

Currently defined robotics are:
  TLD(0)     robot control host = dcap-dca-oradb02

EMM Server = dcap-dca-oradb02

[root@dcap-dcb-oradb02-m bin]# df -k /tapetest
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/cciss/c0d0p8      8254240   7643772    191176  98% /tapetest

[root@dcap-dcb-oradb02-m bin]# df -i /tapetest
Filesystem            Inodes   IUsed   IFree IUse% Mounted on
/dev/cciss/c0d0p8    1048576  217922  830654   21% /tapetest
```

APPENDIX B

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 3.0 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 WAE's were used to provide an endpoint for application optimization.

The GSS's 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 used by the GSS's to communicate:

GSSM-M to GSSs

- TCP ports 2001 - 2009 using a secure session [RMI over SSL (bidirectional)]
- Used for updating configuration changes

GSSs to GSSM-M

- TCP ports 3001 - 3009 configurable range of 30 - 4000 seconds

GSSs to GSSM-M

- UDP:2000
- Auto detect that other GSS's are online

All four GSS's communicate with the CSM's at both data center’s. All four GSS's must be able to reach the CSM's at both DCa and DCb in order to understand their health and availability. This function is called a keepalive. The GSS supports the following keepalives;

- TCP - verify an open/close connection sequence - to termination the sequence, the GSS can be configured to send a RST or a FIN. 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 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

Delegation to GSS Devices

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, that integrates the GSS device into the network's DNS infrastructure and starts delivering user queries to your GSS: modifying your parent domain's DNS server to delegate parts of its name space to your GSS’s. In DCAP 3.0, 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's.

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's were deployed, step by step, from an initial setup perspective.

GSSM-S, GSSM-M, and GSS

There are two GSS's deployed at each data center. Each GSS is connected into each of the two Aggregation switches at each data center. Of the two GSS's at each data center, one is installed as a GSSM and the other is installed as a GSS. A total of four GSS's are installed across both data center A and data center B. Having a GSSM locally at each data center provides for redundancy in the event that

one of the GSSM's 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"

Initial Configuration

The following network settings were configured on all four GSS's in order to allow all the GSS's onto the IP network. Once this initial configuration is complete, all dns configuration is accomplished though the use of a web GUI via HTTPS.

GSSM-M(in DCa)

```
dca-gss-1.gslb.dcap.com
dca-gss-1.gslb.dcap.com#config t
dca-gss-1.gslb.dcap.com(config)#interface ethernet 0
dca-gss-1.gslb.dcap.com(config-eth0)#ip address 101.1.33.10 255.255.255.0
dca-gss-1.gslb.dcap.com(config-eth0)#gss-communications
dca-gss-1.gslb.dcap.com(config-eth0)#duplex full
dca-gss-1.gslb.dcap.com(config-eth0)#speed 100
dca-gss-1.gslb.dcap.com(config-eth0)#exit
dca-gss-1.gslb.dcap.com(config)#hostname dca-gss-1.gslb.dcap.com
Generating certificates
Deploying SSL Web Certificates.
dca-gss-1.gslb.dcap.com(config)#ip default-gateway 101.1.33.1
dca-gss-1.gslb.dcap.com(config)#ip name-server 101.1.33.12
dca-gss-1.gslb.dcap.com(config)#ip name-server 101.1.33.14
dca-gss-1.gslb.dcap.com(config)#ssh enable
dca-gss-1.gslb.dcap.com(config)#telnet enable
dca-gss-1.gslb.dcap.com(config)#ftp enable
dca-gss-1.gslb.dcap.com(config)#ntp enable
dca-gss-1.gslb.dcap.com(config)#ntp-server 101.1.33.12
dca-gss-1.gslb.dcap.com(config)#ntp-server 201.1.33.12
```

GSSM-S(in DCb)

```
dcb-gss-1.gslb.dcap.com
dcb-gss-1.gslb.dcap.com(config)#interface ethernet 0
dcb-gss-1.gslb.dcap.com(config-eth0)#ip address 201.1.33.10 255.255.255.0
dcb-gss-1.gslb.dcap.com(config-eth0)#gss-communications
dcb-gss-1.gslb.dcap.com(config-eth0)#duplex full
dcb-gss-1.gslb.dcap.com(config-eth0)#speed 100
dcb-gss-1.gslb.dcap.com(config-eth0)#exit
dcb-gss-1.gslb.dcap.com(config)#hostname dcb-gss-1.gslb.dcap.com
Generating certificates
Deploying SSL Web Certificates.
dcb-gss-1.gslb.dcap.com(config)#ip default-gateway 201.1.33.1
dcb-gss-1.gslb.dcap.com(config)#ip name-server 201.1.33.14
dcb-gss-1.gslb.dcap.com(config)#ip name-server 201.1.33.12
dcb-gss-1.gslb.dcap.com(config)#telnet enable
dcb-gss-1.gslb.dcap.com(config)#ftp enable
dcb-gss-1.gslb.dcap.com(config)#ntp enable
dcb-gss-1.gslb.dcap.com(config)#ntp-server 101.1.33.12
dcb-gss-1.gslb.dcap.com(config)#ntp-server 201.1.33.12
```

GSS (in DCa)

```
dca-gss-2.gslb.dcap.com(config)#interface ethernet 0
dca-gss-2.gslb.dcap.com(config-eth0)#ip address 101.1.33.11 255.255.255.0
dca-gss-2.gslb.dcap.com(config-eth0)#gss-communications
dca-gss-2.gslb.dcap.com(config-eth0)#duplex full
```

```
dca-gss-2.gslb.dcap.com(config-eth0)#speed 100
dca-gss-2.gslb.dcap.com(config-eth0)#exit
dca-gss-2.gslb.dcap.com(config)#hostname dca-gss-2.gslb.dcap.com
Generating certificates
Deploying SSL Web Certificates.
dca-gss-2.gslb.dcap.com(config)#ip default-gateway 101.1.33.1
dca-gss-2.gslb.dcap.com(config)#ip name-server 101.1.33.14
dca-gss-2.gslb.dcap.com(config)#ip name-server 101.1.33.12
dca-gss-2.gslb.dcap.com(config)#telnet enable
dca-gss-2.gslb.dcap.com(config)#ftp enable
dca-gss-2.gslb.dcap.com(config)#ntp enable
dca-gss-2.gslb.dcap.com(config)#ntp-server 101.1.33.12
dca-gss-2.gslb.dcap.com(config)#ntp-server 201.1.33.12
```

GSS (in DCb)

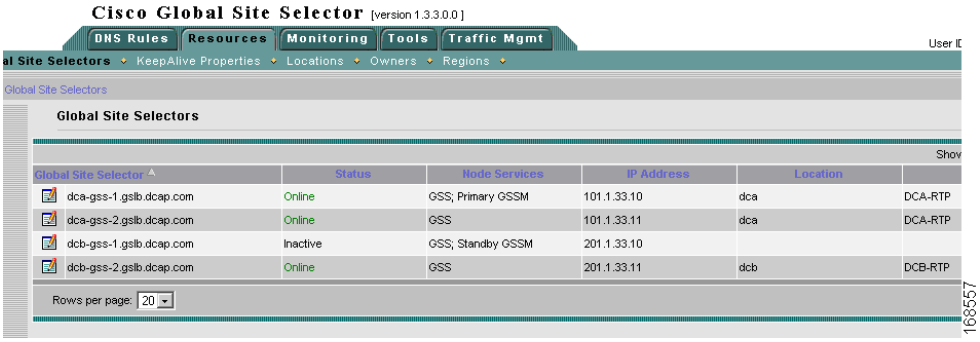
```
dcb-gss-2.gslb.dcap.com(config)#interface ethernet 0
dcb-gss-2.gslb.dcap.com(config-eth0)#ip address 201.1.33.11 255.255.255.0
dcb-gss-2.gslb.dcap.com(config-eth0)#gss-communications
dcb-gss-2.gslb.dcap.com(config-eth0)#duplex full
dcb-gss-2.gslb.dcap.com(config-eth0)#speed 100
dcb-gss-2.gslb.dcap.com(config-eth0)#exit
dcb-gss-2.gslb.dcap.com(config)#hostname dcb-gss-1.gslb.dcap.com
Generating certificates
Deploying SSL Web Certificates.
dcb-gss-2.gslb.dcap.com(config)#ip default-gateway 201.1.33.1
dcb-gss-2.gslb.dcap.com(config)#ip name-server 201.1.33.14
dcb-gss-2.gslb.dcap.com(config)#ip name-server 201.1.33.12
dcb-gss-2.gslb.dcap.com(config)#telnet enable
dcb-gss-2.gslb.dcap.com(config)#ftp enable
dcb-gss-2.gslb.dcap.com(config)#ntp enable
dcb-gss-2.gslb.dcap.com(config)#ntp-server 101.1.33.12
dcb-gss-2.gslb.dcap.com(config)#ntp-server 201.1.33.12
```

DNS Database Configuration Via GSSM-M

After initial IP connectivity to the GSS's was established, the GSSM-M was able to be managed via a web browser using HTTPS. Each GSS must now be added to the GSSM-M via the web browser. This process involves managing the GSSM-M via a browser using HTTPS, clicking on the "Resources" Tab, then the "Global Site Selectors" Tab. At this point, all four GSS's are visible via the GSSM-M. The administrator must then select the appropriate GSS and accept that GSS into the GSS Network. As mentioned earlier up to eight GSS's can be added to an entire GSS Network. In our testing for DCAP 3.0, a total of four GSS's were installed, two at each data center.

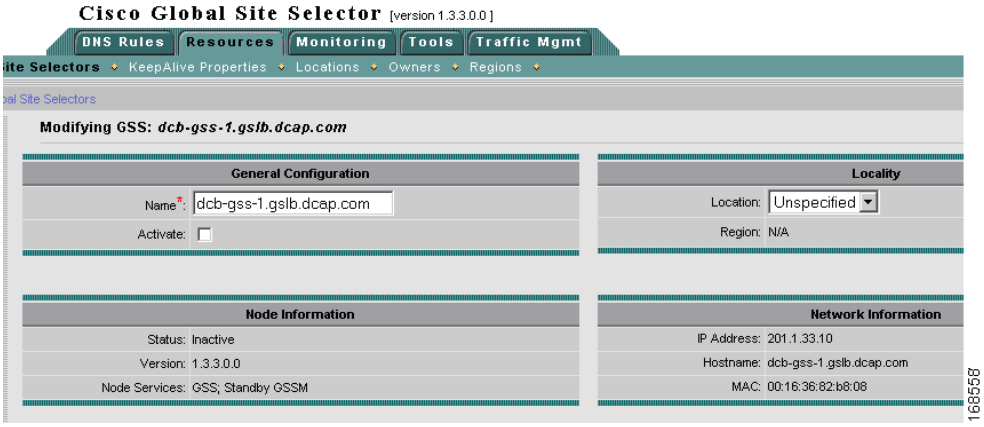
In Figure B-1 the GSS named dcb-gss-1.gslb.dcap.com has not been added to the GSS Network. The following screen capture shows the screen from which the administrator has selected the GSS dcb-gss-1.gslb.dcap.com.

Figure B-1 GSS Selection Screen Capture



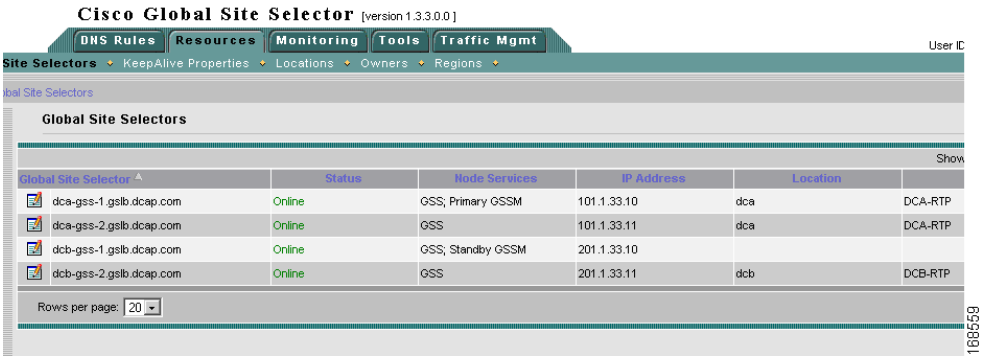
In the "Activate" checkbox, Figure B-2, once the administrator has "activated" this particular GSS, it has joined the GSS network

Figure B-2 GSS Activation Screen Capture



In Figure B-3, we see that the GSS dcb-gss-1.gslb.dcap.com has now joined.

Figure B-3 GSS Join Screen Capture



APPENDIX C

WAAS Implementation

WAAS can be integrated anywhere in the network path. To achieve maximum benefits, optimum placement of the 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.

Design Components

For Phase 3.0 of DCAP testing a dual data center deployment with 3 remote branches was used. Both the functionality of the data center Core and branch edge WAE devices and the redirection router/switches were tested and 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 at the data center/WAN edge for WAAS packet interception
- Cisco WAE-7326 appliance at the data center/WAN edge for aggregation of WAAS services
- Cisco WAE-512 appliance at the data center/Aggregation layer for Central Management
- Cisco 3845, 2821, or 2811 ISR at the branch/remote office for packet interception
- Cisco WAE-612, WAE-512, and/or WAE-502 at the branch/remote office for WAAS termination

WAAS devices in the Core can be placed at either the data center Aggregation Layer or at the WAN edge. Each is suitable and has its benefits and drawbacks. For this deployment the Cisco Enterprise Solutions Engineering (ESE) design with the WAE's deployed at the WAN edge was used.

Data Center Core Details

The Cisco switch providing WAN connectivity for each data center was a Catalyst 6500 running Cisco IOS version 12.2(18)SXF7. A single Wide-Area Application Engine 512 (WAE-512) running Cisco Wide Area Application Service (WAAS) version 4.0.9 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 a single WAE-7326 running Cisco WAAS version 4.0.9 was deployed in application-accelerator mode as a Core WAE and connected to each data center at the WAN edge.

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 version 12.4(12) and 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 version 4.0.9 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 version 12.4(11)T and 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 WAAS 4.0.9 and 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 version 12.4(11)T and was deployed with 69ms and 70ms of latency and bandwidth restricted to that of a T1. A single WAE-502 running WAAS 4.0.9 was installed in the router and connected via the internal backplane connection.

Traffic Redirection Method

The Web Cache Communication Protocol version 2 (WCCPv2) was used as the method of traffic interception and redirection on the data center and branch routers. Generic Routing Encapsulation (GRE) redirection is supported on most router platforms and configured by default. When GRE redirection is deployed, redirected packets will be processed in software. The Catalyst 6500 supports a feature called L2-Redirect which allows all packets to be handled at Layer 2, completely in hardware. Because the Catalyst 6500 uses software to handle a number of other features as well it is recommended that L2 redirect be used when connecting to a Catalyst 6500. This feature must be configured on the WAE, not the Catalyst switch.

Implementation Details

The following sections describes how the WAAS network was deployed.

WAAS Central Manager

A WAE-512 device was connected into each data center in one of the Aggregation Layer swithces. The WAE in DCa was configured as the primary Central Manager and the WAE in DCb was configured as the standby Central Manager. In the event of a DCa failure the standby Central Manager must manually be promoted to the primary role.

Initial Configuration

Once powered up and connected via console, the first-time setup utility was used to enter network settings to give the WAE basic access to the network. (Note: you can re-enter the setup mode by simply typing setup on the CLI if necessary) With the device pingable and the primary-interface specified, the device mode was then configured as central-manager via the CLI. The configuration was saved and then a reload was initiated for the new configuration to take effect.

Primary Central-Manager(in DCA)

```
dca-wae-512-cm#configure terminal
dca-wae-512-cm(config)#
dca-wae-512-cm(config)#primary-interface gigabitEthernet 2/0//Specify inband interface
here
dca-wae-512-cm(config)#interface GigabitEthernet 2/0
dca-wae-512-cm(config-if)#ip address 101.1.33.4 255.255.255.0
dca-wae-512-cm(config-if)#exit
dca-wae-512-cm(config)#device mode central-manager//Designate WAE to run in central
manager mode
dca-wae-512-cm(config)#exit
dca-wae-512-cm#copy run start
dca-wae-512-cm#reload
Proceed with reload?[confirm] y
Shutting down all services, will Reload requested by CLI@ttyS0.
Restarting system.
dca-wae-512-cm#configure terminal
dca-wae-512-cm(config)#central-manager role primary//Designate this WAE as the primary
manager
dca-wae-512-cm(config)#cms enable//Create the Central Management database
dca-wae-512-cm(config)#exit
dca-wae-512-cm#copy run start
```

Standby Central-Manager(in DCb)

```
dca-wae-512-cm#configure terminal
dca-wae-512-cm(config)#
dca-wae-512-cm(config)#primary-interface gigabitEthernet 2/0
dca-wae-512-cm(config)#device mode central-manager
dca-wae-512-cm(config)#exit
dca-wae-512-cm#copy run start
dca-wae-512-cm#reload
Proceed with reload?[confirm] y
Shutting down all services, will Reload requested by CLI@ttyS0.
Restarting system.
dcb-wae-512-cm#configure terminal
dcb-wae-512-cm(config)#central-manager role standby//Designate this WAE as the standby
manager
dcb-wae-512-cm(config)#central-manager address 101.1.33.4//point this WAE to the primary
central-manager
dcb-wae-512-cm(config)#cms enable//register with the primary
dcb-wae-512-cm(config)#exit
dcb-wae-512-cm#copy run start
```

Initial Core WAE Data Center Configuration

A single WAE-7326 in DCA was directly connected to a Catalyst 6500 with a Supervisor 720 running IOS version 12.2(18)SXF7 at the WAN edge of each data center. Similar to the Central Manager, the first-time setup utility was used to configure basic network settings to provide the core device IP

connectivity to the data center network. The interface on the WAE connected to the Catalyst device was configured as the primary interface and the device mode was set to application-accelerator mode. The IP address of the Central Manager was also specified and the core WAE device was then registered to it.

```
dca-wae-7326-1#configure terminal
dca-wae-7326-1(config)#primary-interface gigabitEthernet 2/0
dca-wae-7326-1(config)#device mode application-accelerator
dca-wae-7326-1(config)#central manager address 101.1.33.4
dca-wae-7326-1(config)#cms enable
Registering WAAS Application Engine...
Sending device registration request to Central Manager with address 101.1.33.4
Please wait, initializing CMS tables
Successfully initialized CMS tables
Registration complete.
Please preserve running configuration using 'copy running-config startup-config'.
Otherwise management service will not be started on reload and node will be shown
'offline' in WAAS Central Manager UI.
management services enabled
dca-wae-7326-1(config)#end
dca-wae-7326-1#copy run start
dca-wae-7326-1#
```

Note IP connectivity to the Central Manager must first be established before registering.

The WAE devices have two Gigabit Ethernet interfaces that can be used to connect into the network. Often times one of the two interfaces is used for out of band management however it can be used as a redundant link to the WCCP switch. The set up in DCb made use of the standby interface on the WAE device which was configured and used for in-band redundancy.

```
dcb-wae-7326-1#configure terminal
dcb-wae-7326-1(config)#interface Standby 1//configure the Standby interface
dcb-wae-7326-1(config-if)#description standby interface group
dcb-wae-7326-1(config-if)#ip address 10.0.14.3 255.255.255.0//give it an IP address
dcb-wae-7326-1(config)#interface GigabitEthernet 1/0
dcb-wae-7326-1(config-if)#description dcb-wan-1 gi1/8
dcb-wae-7326-1(config-if)#standby 1//assign the physical interface to the standby group
dcb-wae-7326-1(config-if)#exit//the default priority is 100
dcb-wae-7326-1(config)#interface GigabitEthernet 2/0
dcb-wae-7326-1(config-if)#description dcb-wan-1 gi1/3
dcb-wae-7326-1(config-if)#standby 1 priority 105//assign the physical interface to the
standby group
dcb-wae-7326-1(config-if)#exit//with a higher than default priority
dcb-wae-7326-1(config)#primary-interface Standby 1//specify the Standby interface as the
primary interface
dcb-wae-7326-1# cms enable
Registering WAAS Application Engine...
Sending device registration request to Central Manager with address 101.1.33.4
Please wait, initializing CMS tables
Successfully initialized CMS tables
Registration complete.
Please preserve running configuration using 'copy running-config startup-config'.
Otherwise management service will not be started on reload and node will be shown
'offline' in WAAS Central Manager UI.
management services enabled
dca-wae-7326-1(config)#end
dca-wae-7326-1#copy run start
dca-wae-7326-1#
```

Initial Edge WAE Remote Branch Configuration

Branch 1

Two Cisco WAE's, a WAE-512 and a WAE-612, were connected to a Cisco 3845 ISR running IOS version 12.4(12) at remote Branch 1. The edge WAE devices were brought online and the first-time setup utility was used to configure basic network settings to provide the edge device IP connectivity to the branch LAN. Once complete the interface on the WAE connecting to the ISR was configured as the primary interface and the device mode was set to application-accelerator mode. The IP address of the Central Manager was also specified and the edge WAE device was registered with it as done with the Core devices.

Branch 2

A NM-WAE-502 was inserted into a Cisco 2821 ISR running IOS version 12.4(11)T at remote Branch 2. The T train code had to be used because non T train code did not support the NM WAE hardware at the time this testing was done. Secondly, a Cisco WAE-512 was connected to the router via Gigabit Ethernet copper. The edge WAE devices were brought online, configured for IP connectivity, and registered with the Central Manager.

Branch3

A single NM-WAE-502 was inserted into a Cisco 2811 ISR running IOS version 12.4(11)T at remote Branch 3. Like at Branch 2, the T train code had to be used at this branch as well.

WAN Connection

For Phase 3.0, Netem software, which is built into the 2.6 and later linux kernel, that provides network emulation functionality for testing protocols by emulating properties of wide-area networks, was used to emulate and simulate different WAN latencies and bandwidths required for each of the three remote branches. A single dual-core Penguin server equipped with 6 Gigabit Ethernet interfaces running Cisco Enterprise Linux 5.0 was used to connect the 3 branches and 2 data centers together. A single port from each of the data center WAN devices, dca-wan-1 and dcb-wan-1, was connected to the server. Similarly, a single port from each of the 3 branch routers connected to the server. The Netem software is able to introduce delay and bandwidth at Layer 2. The **vconfig** command was used to create VLANS which were later bridged together appropriately with several commands from the bridge utilities package. This allowed packets to flow between the Ethernet interfaces. Once the interfaces were bridged appropriately the **tc qdisc** Netem software commands were used to add the appropriate latency and bandwidth restrictions.

Bandwidth/Latency to DCA/Latency to DCb

- Branch 1: T3/5ms/6ms
- Branch 2: T1/17ms/18ms
- Branch 3: T1/69ms/70ms

WAAS Network Configuration Via the Central Manager

After the initial IP connectivity to the WAAS Central Manager is established the Central Manager GUI was then reachable by opening a secure web browser (IE is the only browser supported) to the IP address on port 8443. (ie. https://101.1.33.4:8443/) It was verified in this Central Manager GUI that the registration actions were in fact successful. Once the devices have been registered with the CMS enable

command they stay in a Pending state until you activate each device through the GUI. Once you have activated the devices the CMS status for each device will change to be online. Both the CLI and GUI can be used for configuring the WAAS network. The GUI was used in this deployment and the following instructions use the GUI as the main method for configuration.

Configure Device Groups

It is easiest to configure groups of devices at once rather than one at a time. The best way to group devices together is to determine which ones will need mirroring, or closely mirroring, configurations. In this network three main Device groups were set up. By default the AllDevicesGroup was created which contained all devices that have been registered with the Central Manager. After this two new groups were created, one for the Core devices and one for the Edge devices. Create these by navigating/configuring as follows:

Step 1 Create new group|Name (fill in)|Submit.

Once the device group is created, select it from the main Device Group's GUI page and click the Assign Devices on the left sidebar. Select the appropriate devices by clicking the blue "X" next to the device name. Once all the devices for the group have been selected click the Submit button.

Core Cluster Settings

A core cluster was first defined which consisted of the single file-server, running Windows Server 2003, in the data center used for centralized file storage. The cluster was defined by navigating/configuring through the GUI as follows:

Step 1 Navigate to Devices|Device Groups|Create New Device Group.

Step 2 Enter the following values:

- Type: WAFS Core Cluster
- Name: WAFS Core Cluster

Once defined, Core Server settings can be specified.

Step 3 Navigate to Devices|Device Groups|Core Server Settings.

Step 4 Enter the appropriate values.

- File server access username/password

Configure WAE Devices for Domain Name System (DNS)

The DNS device used for testing was a Windows Server 2003 server, Server-2k3, located in the data center. To configure the WAE's for DNS navigate through the GUI as follows:

Step 1 Navigate to Devices|Device Groups|AllDevicesGroup|General Settings|Network|DNS.

Step 2 Enter the following values:

- Local Domain Name: dcap.com

- List of DNS Servers: 101.1.33.12 101.1.33.14 201.1.33.12

Step 3 Submit and force setting on all devices in group.

Configure WAE Devices for Windows Name Services (WINS)

WINS can be used instead of DNS in some circumstance like when no DNS is available. WINS, as well as DNS, was configured on Server-2k3 in the data center. The WINS settings were configured to WAE devices by navigating/configuring through the GUI as follows:

Step 1 Navigate to Devices|Device Groups|AllDevicesGroup|General Settings|Windows Name Services.

Step 2 Enter the following values:

- Workgroup or Domain Name: DCAP
- WINS Server: 101.1.33.12

Step 3 Navigate to Devices|Devices|WAE Device|(Advanced)Activation to configure (or verify) NETBIOS names for the WAE devices.

Step 4 Enter the appropriate value.

- NetBIOS Name: <name>

Configure NTP on the Central Manager

The lab NTP server used is running on goel.cisco.com (172.18.177.132). This server was used to sync the Central Manager with the rest of the data center devices. The configuration was done via the GUI by navigating/configuring as follows:

Step 1 Navigate to Devices|Central Manager (primary)|Show Advanced|General Settings|Miscellaneous|Date/Time|NTP/Time Zone.

Step 2 Enter the following values:

- Time Zone: EST
- NTP: Check Enable Box
- NTP Server: 172.18.177.13

Configure NTP on Core and Edge WAE Devices

The Core and Edge devices were then synchronized to the Central Manager by navigating/configuring through the GUI as follows:

Step 1 Navigate to Devices|Device Groups|AllDevicesGroup|General Settings|Miscellaneous|Date/Time|NTP/Time Zone.

Step 2 Enter the following values:

- Time Zone: EST
- NTP: Check Enable Box
- NTP Server: 10.0.13.6(CM address)

Defining the Core WAE

The core WAE-7326 device was defined via the GUI as a Core device by navigating as follows:

Step 1 Navigate to Devices|Devices|Select device to define as Core WAE|File Services|Core Configuration.

Step 2 Check enable Core Server.

Step 3 Select predefined Core WAFS Cluster.

The device reloaded as instructed.

Step 4 Navigate to Devices|Device Groups|Core Cluster|Members|Devices|Devices.

Step 5 Verify the device is now Core and CMS status is online.

Defining the Edge WAE

The edge WAE-512, WAE-612 and WAE-502 devices were defined via the GUI as an Edge device by navigating as follows:

Step 1 Navigate to Devices|Device Groups|Select Edge device group|File Services|Edge Configuration

Step 2 Enable Edge Server and leave all boxes checked.

Step 3 Select predefined Core WAFS Cluster.

The devices reloaded as instructed.

Step 4 Navigate to Devices|Devices.

Step 5 Verify the device is now Edge and CMS status is online.

Configure WAE Authentication Methods

With the addition of CIFS Auto-discovery, each WAE no longer must be authenticated with the Windows File Server before CIFS optimizations will occur. The WAE devices previously configured for Windows are now simply configured to authenticate locally.

Step 1 Navigate to Devices|Device Groups|AllDevicesGroup|General Settings|Authentication|Authentication Methods

Step 2 Check Authentication Login Methods and Authorization Methods, and set Local to the primary login method.

Configure a File Server

With the addition of CIFS Auto-discovery it is no longer necessary to define specific file servers.

Create a New Connection

A connection was then established between Core and Edge WAE devices by navigating/configuring through the GUI as follows:

Step 1 Navigate to Services|File|Connectivity|Definition.

Step 2 Enter the following values:

- Name: wae-connection-1

Step 3 Assign Core Cluster and submit

Step 4 Navigate to Services|File|Connectivity|Definition|Assign Edge Devices.

Step 5 Click the blue X next to each Edge device, then click Submit.

Basic Server/Client Configuration Overview

A server in each data center was installed with Windows Server 2003 SP4. Active directory, DNS, WINS and Internet Information Services (IIS) was configured on the servers and the domain used was dcap.com. Three client PC's were used at each branch location, one running Windows XP, one running Linux, and one running Windows 2003. The Windows 2003 server was acting as a local Domain Controller and local DNS server at each branch.

Note Cisco WAAS is not able to apply the full breadth of CIFS acceleration when a feature known as "digital signatures" are enabled on the file server (default for Windows 2000 SP4 and Windows 2003 SP1 and above).

To disable digital signatures, login to a domain controller and open the "Default Domain Controller Security Settings" MMC snap-in, generally found under Start > Programs > Administrative Tools. From the left pane, expand to Security Settings > Local Policies > Security Options, and locate a policy called:

- Domain member: Digitally encrypt or sign secure channel data (always) - double-click this item, ensure that "Define this policy setting" is checked, and that the radio button next to "disabled" is selected.
- Microsoft network server: Digitally sign communications (always) - double-click this item, ensure that "Define this policy setting" is checked, and that the radio button next to "disabled" is selected.
- Microsoft network server: Digitally sign communications (if client agrees) - double-click this item, ensure that "Define this policy setting" is checked, and that the radio button next to "disabled" is selected.

Reboot server, or execute the **gpupdate /force** command from a DOS window to refresh policy settings.

WCCPv2 Overview

Cisco WAAS utilizes a WCCPv2 service group called “TCP Promiscuous.” The TCP promiscuous service group is actually a combination of two service groups: 61 and 62.

While both service groups look for the same traffic, only one service group should be in the path for each direction of traffic flow. If service group 61 is configured in such a way that it intercepts traffic going from the client to the server, service group 62 should be configured to intercept traffic returning to the client from the server. Given that the service groups distribute load based on the source IP or destination IP of the packets, this ensures that the packets will be routed to the same WAE regardless of the direction of traffic flow.

Service group 61 redirects TCP packets to a WAE device, distributes load based on the source IP address of the flow.

Service group 62 redirects TCP packets to a WAE device, distributes load based on the destination IP address of the flow.

When the TCP promiscuous mode service is used, the CIFS caching service is not required. (WCCP Version 2 service 89).

WCCPv2 Implementation

The DCa and DCb WAN routers, as well as each of the three branch ISR’s, were configured to use WCCPv2 as the method for traffic interception and redirection. On the data center WAN Supervisor 720 router, all interfaces from the downstream core switches were configured to redirect ingress traffic using WCCPv2 service group 62. For DCAP 3.0 a single Gigabit Ethernet port was used to connect to the Linux WAN emulator. The WAN port was configured to redirect ingress traffic using WCCPv2 service group 61. Each of the three remote Branch ISR’s were configured with service group 61 on the ingress LAN port and service group 61 on its ingress WAN port. The service group deployment scenario used is the most commonly deployed scenario.

Note The service group deployment at the branch and data center are opposite, however, they are functionally equivalent.

Testing Concept

DCAP 3.0 WAAS testing focused on the basic network deployment, functionality and configuration for the WAE devices. 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 true HTTP and FTP traffic. Performance improvements were evaluated during the testing, however, no stress testing was performed.

APPENDIX D

Blade Server Deployment

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 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. Four Cisco 3020 switches were installed in each enclosure and the BladeSystem's were 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 only one of these switches was used to provide dual-homed connectivity to the data center LAN aggregation switches. This was because this phase of testing was focused on providing the basic functionality and connectivity to the data center networks. No NIC teaming/bonding was used anywhere in the topology so no redundancy of Cisco 3020 switches was an option for this phase. Typically a more redundant deployment would be to team NIC's on servers and connect multiple L2 blade switches to each of the Aggregation switches. This not only provides redundancy but also provides increased bandwidth. This will be a focus in future testing in the DCAP network.

Initial Configuration of the 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. 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 has been set for each of the devices in the system, configuration and management of all devices can be done through the HP OA GUI.

Initial Configuration of the 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.

Note If using Windows 2003 Server, make sure the security setting are set to allow this connection to be made on a non-secure HTTP page.

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.

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.

Configuring the 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, 4-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. Since the switches were already wired previously in the setup, the only steps left to provide network connectivity is to configure the 4- port Portchannels between the layer 2+3 3020 and layer 2+3 6500 switches and configure the downlink to server switch ports. 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. For this deployment all Windows installations mapped directly to the switches as you would expect. (NIC 1 to Switch in slot 1, NIC 2 to Switch in slot 2, etc.) This was not the case for the Linux installations in that the NIC to switch mappings were one off. (NIC 1 to Switch 2, NIC 2 to Switch 3, etc) To correct this, the HW MAC addresses in the Linux network configuration were shifted to make the correlation 1:1. Bringing down the Linux port or internal switch port and checking the link status on the Linux server with the ethtool command was helpful in determining the mappings within the chassis. 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 which the server is to exist on.

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 Applications Configuration Details

This appendix details the DCAP Oracle E-Business Suite environment software and hardware configurations. An Oracle Vision Demo environment was installed in multi-node mode using the Oracle installation tool “Rapid Install.” 11.5.10.2 in installed by default with database verion 9iR2. The DB is upgraded to Oracle 10gR2 and Oracle Applications are upgraded to latest Technology stack ATG.RUP4 patchset. The Application Tier and the Database Tier have been enabled with Autoconfig.

- Application URL: http://wwwin-oefin.gslb.dcap.com
- HTTP Port : 8000
- Forms Port : 9000

Application Configuration

The following data center Application hosts are used.

DCa

- Dcap-dca-oraapp01 (Web & Forms)
- Dcap-dca-oraapp02 (Web & Forms)
- Dcap-dca-oracm01 (Concurrent Manager)

DCb

- Dcap-dcb-oraapp01(Web & Forms)
- Dcap-dcb-oraapp02(Web & Forms)
- Dcap-dcb-oracm01(Web & Forms)

Key configuration files crucial to Oracle E-business Application functions are:

- Application Context file (Eg: OEFIN_dcap-dca-oraapp01.xml)
- Tnsnames.ora & Listener.ora files for Oracle 8.0.6 and iAS products
- Environment file that sets the appropriate variables.(OEFIN_dcap-dca-oraapp01.env).

Configuration files from application host dcap-dca-oraapp01 are listed for reference.

Application Context file


```
<?xml version = '1.0'?>
<!-- $Header: adxmlctx.tmp 115.412 2006/11/02 12:35:25 pvaddepa ship $ -->
<!-- ################################################################

  This file is automatically generated by AutoConfig.  It will be read and
  overwritten.  If you were instructed to edit this file, or if you are not
  able to use the settings created by AutoConfig, refer to Metalink document
  165195.1 for assistance.

  ################################################################ -->
<oa_context version="$Revision: 115.412 $">
   <oa_context_name oa_var="s_contextname">OEFIN_dcap-dca-oraapp01</oa_context_name>
   <oa_context_serial oa_var="s_contextserial">3</oa_context_serial>
   <oa_context_type oa_var="s_contexttype">APPL_TOP Context</oa_context_type>
   <oa_context_file_loc
oa_var="s_contextfile">/apps/oefin/appl_top/admin/OEFIN_dcap-dca-oraapp01.xml</oa_context_
file_loc>
   <oa_system>
      <oa_system_name oa_var="s_systemname">OEFIN</oa_system_name>
      <global_db_name oa_var="s_dbSid">OEFIN</global_db_name>
      <global_db_name oa_var="s_dbGlnam">OEFIN</global_db_name>
      <db_name_lower oa_var="s_dbSidLower">oefin</db_name_lower>
      <PRINTER oa_var="s_printer">noprint</PRINTER>
      <oa_system_config>
         <TIER_DB oa_var="s_isDB">NO</TIER_DB>
         <TIER_ADMIN oa_var="s_isAdmin">NO</TIER_ADMIN>
         <TIER_WEB oa_var="s_isWeb">YES</TIER_WEB>
         <TIER_FORMS oa_var="s_isForms">YES</TIER_FORMS>
         <TIER_NODE oa_var="s_isConc">NO</TIER_NODE>
         <TIER_FORMSDEV oa_var="s_isFormsDev">YES</TIER_FORMSDEV>
         <TIER_NODEDEV oa_var="s_isConcDev">NO</TIER_NODEDEV>
         <TIER_WEBDEV oa_var="s_isWebDev">YES</TIER_WEBDEV>
         <config_option type="techstack" oa_var="s_techstack">ias1022</config_option>
         <config_option type="techstack" oa_var="s_tnsmode">generateTNS</config_option>
         <config_option type="imeetingstack"
oa_var="s_imeeting_stack">imeetingNotInstalled</config_option>
         <config_option type="adx" oa_var="s_apps_version">11.5.10.2</config_option>
         <PORT_POOL oa_var="s_port_pool">-1</PORT_POOL>
      </oa_system_config>
      <oa_users>
         <oa_user type="SYS">
            <username oa_var="s_sys_user">SYS</username>
         </oa_user>
         <oa_user type="APPS">
            <username oa_var="s_apps_user">APPS</username>
            <username oa_var="s_applsys_user">APPLSYS</username>
         </oa_user>
         <oa_user type="GUEST">
            <username oa_var="s_guest_user">GUEST</username>
            <password oa_var="s_guest_pass">ORACLE</password>
         </oa_user>
         <oa_user type="GWYUID">
            <username oa_var="s_gwyuid_user">APPLSYSPUB</username>
            <password oa_var="s_gwyuid_pass">PUB</password>
         </oa_user>
      </oa_users>
      <nls_settings>
         <defterr oa_var="s_defterr">AMERICA</defterr>
         <base_lang oa_var="s_base_lang">US</base_lang>
         <env_langs oa_var="s_env_langs">US</env_langs>
      </nls_settings>
```

```
      <oa_db_server>
         <dbhost oa_var="s_dbhost">db-oefin</dbhost>
         <domain oa_var="s_dbdomain">gslb.dcap.com</domain>
         <dbsid oa_var="s_db_serv_sid">OEFIN</dbsid>
         <dbport oa_var="s_dbport" oa_type="PORT">1521</dbport>
         <dbcset oa_var="s_dbcset">UTF8</dbcset>
      </oa_db_server>
      <oa_admin_server>
         <hostname oa_var="s_admhost">dcap-dca-oraapp01</hostname>
         <domain oa_var="s_admdomain">dcap.com</domain>
         <admin_ui_access_nodes
oa_var="s_admin_ui_access_nodes">localhost</admin_ui_access_nodes>
      </oa_admin_server>
      <oa_cp_server>
         <hostname oa_var="s_cphost">dcap-dca-oraapp01</hostname>
         <domain oa_var="s_cpdomain">dcap.com</domain>
         <reports_port oa_var="s_repsport" oa_type="PORT">7000</reports_port>
         <rpc_port oa_var="s_rpcport" oa_type="PORT">1626</rpc_port>
         <conc_java_ldlib oa_var="s_conc_java_ldlib"
osd="UNIX">/apps/oefin/product/iAS/lib:/apps/oefin/product/8.0.6/network/jre11/lib/i686/na
tive_threads:/apps/oefin/product/8.0.6/network/jre11/lib/linux/native_threads:/apps/oefin/
appl_top/cz/11.5.0/bin:${LD_LIBRARY_PATH:=}</conc_java_ldlib>
         <cp_reviver oa_var="s_cp_reviver">disabled</cp_reviver>
         <fndreviverpiddir
oa_var="s_fndreviverpiddir">/apps/oefin/appl_top/fnd/11.5.0/log</fndreviverpiddir>
      </oa_cp_server>
      <oa_workflow_server>
         <hostname oa_var="s_wfhost">%s_wfhost%</hostname>
         <domain oa_var="s_wfdomain">dcap.com</domain>
         <hostname oa_var="s_javamailer_imaphost">NoImapHost</hostname>
         <domain oa_var="s_javamailer_imapdomainname">NoImapDomain</domain>
         <username oa_var="s_wf_admin_role">SYSADMIN</username>
         <username oa_var="s_javamailer_reply_to">NoReplyTo</username>
         <username oa_var="s_javamailer_imap_user">NoImapUser</username>
      </oa_workflow_server>
      <oa_smtp_server>
         <hostname oa_var="s_smtphost">dcap-dca-oraapp01</hostname>
         <domain oa_var="s_smtpdomainname">dcap.com</domain>
      </oa_smtp_server>
      <oa_forms_server>
         <hostname oa_var="s_formshost">dcap-dca-oraapp01</hostname>
         <domain oa_var="s_formsdomain">dcap.com</domain>
         <formsfndtop oa_var="s_formsfndtop">/apps/oefin/appl_top/fnd/11.5.0</formsfndtop>
         <forms_port oa_var="s_formsport" oa_type="PORT">9000</forms_port>
         <forms_runtime oa_var="s_frmRuntime" osd="UNIX">f60webmx</forms_runtime>
         <forms_runcgi oa_var="s_frmRunCGI" osd="UNIX">f60cgi</forms_runcgi>
         <forms_connect oa_var="s_frmConnectMode">socket</forms_connect>
         <forms_walletdir
oa_var="s_frmWalletDir">/apps/local/OEFIN/8.0.6/forms60/wallet</forms_walletdir>
         <tcf_port oa_var="s_tcfport" oa_type="PORT">-1</tcf_port>
         <desformat oa_var="s_desformat">html</desformat>
         <forms_network_retries oa_var="s_frmNetworkRetries">0</forms_network_retries>
         <chronos_enabled oa_var="s_isChronosEnabled"/>
         <chronosURL
oa_var="s_chronosURL">http://dcap-dca-oraapp01.dcap.com:8000/oracle_smp_chronos/oracle_smp
_chronos_sdk.gif</chronosURL>
         <forms_trace_path
oa_var="s_forms_trace_path">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01</form
s_trace_path>
         <forms_jvm_options oa_var="s_forms_jvm_options" osd="Linux">-Xmx256M -Xms128M
-XX:NewSize=60M -XX:MaxPermSize=128M -XX:MaxNewSize=120M -Xrs</forms_jvm_options>
         <forms_heartbeat oa_var="s_forms_heartbeat">2</forms_heartbeat>
      </oa_forms_server>
      <oa_met_server>
```

```
            <hostname oa_var="s_methost">dcap-dca-oraapp01</hostname>
            <domain oa_var="s_metdomain">dcap.com</domain>
            <met_data_port oa_var="s_metdataport" oa_type="PORT">9100</met_data_port>
            <met_req_port oa_var="s_metreqport" oa_type="PORT">9200</met_req_port>
            <met_least_loaded oa_var="s_leastloadedhost">dcap-dca-oraapp01</met_least_loaded>
            <met_error_url oa_var="s_meterrorurl"/>
         </oa_met_server>
         <oa_mwa_server>
            <hostname oa_var="s_mwahost">dcap-dca-oraapp01</hostname>
            <domain oa_var="s_mwadomain">dcap.com</domain>
            <mwaLogLevel oa_var="s_mwaLogLevel">error</mwaLogLevel>
            <mwaLogRotate oa_var="s_mwaLogRotate">Yes</mwaLogRotate>
            <mwaLogFileSize oa_var="s_mwaLogFileSize">10000000</mwaLogFileSize>
            <mwaDropConnectionTimeout
oa_var="s_mwaDropConnectionTimeout">5</mwaDropConnectionTimeout>
            <mwaStaleSessionTimeout
oa_var="s_mwaStaleSessionTimeout">60</mwaStaleSessionTimeout>
            <mwaDispatcherThreadCount
oa_var="s_mwaDispatcherThreadCount">15</mwaDispatcherThreadCount>
            <mwaDispatcherClientsPerWorker
oa_var="s_mwaDispatcherClientsPerWorker">10</mwaDispatcherClientsPerWorker>
            <mwaJVMb oa_var="s_mwaJVMb">FALSE</mwaJVMb>
            <mwaPortNo oa_var="s_mwaPortNo" oa_type="PORT">10200</mwaPortNo>
            <mwaTelnetPortNo oa_var="s_mwaTelnetPortNo"
oa_type="PORT">10200</mwaTelnetPortNo>
            <mwaDispatcherPort oa_var="s_mwaDispatcherPort"
oa_type="PORT">10800</mwaDispatcherPort>
            <mwaActivateLOVByEnter
oa_var="s_mwaActivateLOVByEnter">FALSE</mwaActivateLOVByEnter>
            <mwaSubmenuChangeOrgResp
oa_var="s_mwaSubmenuChangeOrgResp">FALSE</mwaSubmenuChangeOrgResp>
         </oa_mwa_server>
         <oa_disco_server>
            <disco_portrange oa_var="s_disco_servlet_portrange"
oa_type="PORT">17000-17009</disco_portrange>
            <disco_ver_comma oa_var="s_disco_ver_comma">4,1,48,08</disco_ver_comma>
            <oem_web_port oa_var="s_oemweb_port" oa_type="PORT">10000</oem_web_port>
            <osagent_port oa_var="s_osagent_port" oa_type="PORT">10100</osagent_port>
            <disco_nprocs oa_var="s_disco_nprocs" osd="Linux">1</disco_nprocs>
            <disco_eul_prefix oa_var="s_disco_eul_prefix">EUL</disco_eul_prefix>
            <disco_node_weight oa_var="s_disco_node_weight" osd="Linux">1</disco_node_weight>
            <discoinstancename
oa_var="s_discoinstance">dcap-dca-oraapp01.dcap.com_8000</discoinstancename>
            <disco_standalone oa_var="s_disco_standalone">false</disco_standalone>
            <disco_protocol oa_var="s_disco_protocol">http</disco_protocol>
            <disco_machine oa_var="s_disco_machine">wwwin-oefin.gslb.dcap.com</disco_machine>
            <disco_port oa_var="s_disco_port" oa_type="PORT">8000</disco_port>
            <disco_ipaddress oa_var="s_disco_ip_addr"/>
            <disco_ORBalwaysProxy oa_var="s_disco_ORBalwaysProxy">no</disco_ORBalwaysProxy>
            <disco_jretop oa_var="s_disco_jretop"
osd="Linux">/apps/oefin/product/8.0.6/jre1183o</disco_jretop>
            <disco_jvm_options oa_var="s_disco_jvm_options" osd="unix"/>
            <disco_oad_executable oa_var="s_disco_oad_executable">oad</disco_oad_executable>
         </oa_disco_server>
         <oa_reports_server>
            <rptsminengine oa_var="s_minengine">0</rptsminengine>
            <rptsmaxengine oa_var="s_maxengine">1</rptsmaxengine>
         </oa_reports_server>
         <oa_web_server>
            <hostname oa_var="s_webhost">dcap-dca-oraapp01</hostname>
            <externURL
oa_var="s_external_url">http://wwwin-oefin.gslb.dcap.com:8000</externURL>
            <directory_index oa_var="s_directory_index">index.html</directory_index>
            <webentryhost oa_var="s_webentryhost">wwwin-oefin</webentryhost>
```

```
        <webentrydomain oa_var="s_webentrydomain">gslb.dcap.com</webentrydomain>
        <lock_pid_dir
oa_var="s_lock_pid_dir">/apps/local/OEFIN/iAS/Apache/Apache/logs</lock_pid_dir>
        <domain oa_var="s_webdomain">dcap.com</domain>
        <server_ip_address oa_var="s_server_ip_address"/>
        <web_port oa_var="s_webport" oa_type="PORT">8000</web_port>
        <web_port_pls oa_var="s_webport_pls" oa_type="PORT">8200</web_port_pls>
        <oprocmgr_port oa_var="s_oprocmgr_port" oa_type="PORT">8100</oprocmgr_port>
        <activewebport oa_var="s_active_webport" oa_type="PORT">8000</activewebport>
        <servlet_portrange oa_var="s_forms_servlet_portrange"
oa_type="PORT">18000-18009</servlet_portrange>
        <oacore_portrange oa_var="s_oacore_servlet_portrange"
oa_type="PORT">16000-16009</oacore_portrange>
        <servlet_port oa_var="s_servletport" oa_type="PORT">8800</servlet_port>
        <jtf_fulfillment_port oa_var="s_jtfuf_port"
oa_type="PORT">9300</jtf_fulfillment_port>
        <imtsrvport oa_var="s_imtsrvport" oa_type="PORT">9500</imtsrvport>
        <imtrecport oa_var="s_imtrecport" oa_type="PORT">9600</imtrecport>
        <imtimonport oa_var="s_imtimonport" oa_type="PORT">9700</imtimonport>
        <mapviewerport oa_var="s_mapviewer_port" oa_type="PORT">9800</mapviewerport>
        <web_pid_file
oa_var="s_web_pid_file">/apps/local/OEFIN/iAS/Apache/Apache/logs/httpd.pid</web_pid_file>
        <oacore_trusted_oproc_nodes
oa_var="s_oacore_trusted_oproc_nodes">dcap-dca-oraapp01</oacore_trusted_oproc_nodes>
        <enable_trusted_nodes_access oa_var="s_enable_trusted_nodes_access"/>
        <trusted_admin_client_nodes
oa_var="s_trusted_admin_client_nodes">dcap-dca-oraapp01</trusted_admin_client_nodes>
        <dad_stateful_option
oa_var="s_dad_stateful_option">STATELESS_RESET</dad_stateful_option>
        <jserv_classpath_begin oa_var="s_jserv_classpath_begin"># wrapper.classpath
=</jserv_classpath_begin>
        <jserv_classpath_end oa_var="s_jserv_classpath_end"># wrapper.classpath
=</jserv_classpath_end>
        <jserv_zone_classpath_begin oa_var="s_jserv_zone_classpath_begin"/>
        <jserv_zone_classpath_end oa_var="s_jserv_zone_classpath_end"/>
        <jvm_options oa_var="s_jvm_options" osd="Linux">-verbose:gc -Xmx512M -Xms128M
-XX:MaxPermSize=128M -XX:NewRatio=2 -XX:+PrintGCTimeStamps -XX:+UseTLAB</jvm_options>
        <jvm_startup oa_var="s_jvm_startup" osd="UNIX">java</jvm_startup>
        <jservconf oa_var="s_jservconf">jserv.conf</jservconf>
        <jserv_secret_key
oa_var="s_jserv_secret_key">/apps/local/OEFIN/iAS/Apache/Jserv/conf/jserv.secret.key</jser
v_secret_key>
        <jserv_server_authentication
oa_var="s_jserv_server_authentication">true</jserv_server_authentication>
        <formservlet_maxblocktime
oa_var="s_forms_maxblocktime">5000</formservlet_maxblocktime>
        <server_url oa_var="s_forms_servlet_serverurl"/>
        <servlet_comment oa_var="s_forms_servlet_comment">#</servlet_comment>
        <formservlet_session_cookie
oa_var="s_form_session_cookie">true</formservlet_session_cookie>
        <url_protocol oa_var="s_url_protocol">http</url_protocol>
        <local_url_protocol oa_var="s_local_url_protocol">http</local_url_protocol>
        <web_ssl_port oa_var="s_webssl_port" oa_type="PORT">443</web_ssl_port>
        <web_ssl_directory
oa_var="s_web_ssl_directory">/apps/local/OEFIN/iAS/Apache/Apache/conf</web_ssl_directory>
        <web_ssl_keyfile
oa_var="s_web_ssl_keyfile">/apps/local/OEFIN/iAS/Apache/Apache/conf/ssl.key/server.key</we
b_ssl_keyfile>
        <web_ssl_certfile
oa_var="s_web_ssl_certfile">/apps/local/OEFIN/iAS/Apache/Apache/conf/ssl.crt/server.crt</w
eb_ssl_certfile>
        <web_ssl_certchainfile
oa_var="s_web_ssl_certchainfile">/apps/local/OEFIN/iAS/Apache/Apache/conf/ssl.crt/ca-bundl
e.crt</web_ssl_certchainfile>
```

```
        <websrvwallet
oa_var="s_websrv_wallet_file">/apps/local/OEFIN/iAS/Apache/Apache/conf</websrvwallet>
        <sun_plugin_ver oa_var="s_sun_plugin_ver">1.4.2_04</sun_plugin_ver>
        <sun_plugin_type oa_var="s_sun_plugin_type">jinit</sun_plugin_type>
        <sun_clsid oa_var="s_sun_clsid">CAFEEFAC-0014-0002-0004-ABCDEFFEDCBA</sun_clsid>
        <jinit_ver_dot oa_var="s_jinit_ver_dot">1.3.1.21</jinit_ver_dot>
        <jinit_ver_comma oa_var="s_jinit_ver_comma">1,3,1,21</jinit_ver_comma>
        <jinit_clsid
oa_var="s_jinit_clsid">CAFECAFE-0013-0001-0021-ABCDEFABCDEF</jinit_clsid>
        <proxyhost oa_var="s_proxyhost"/>
        <proxyport oa_var="s_proxyport"/>
        <proxybypassdomain oa_var="s_proxybypassdomain">dcap.com</proxybypassdomain>
        <portal30_user oa_var="s_portal30_user">portal30</portal30_user>
        <portal30_sso_user oa_var="s_portal30_sso_user">portal30_sso</portal30_sso_user>
        <portal30_ppesslpath
oa_var="s_portal30_ppesslpath">changeOnPortalConfiguration</portal30_ppesslpath>
        <portal30_dad oa_var="s_portal30_dad">OEFIN_portal30</portal30_dad>
        <portal30_sso_dad
oa_var="s_portal30_sso_dad">OEFIN_portal30_sso</portal30_sso_dad>
        <apps_portal_url
oa_var="s_apps_portal_url">http://dcap-dca-oraapp01.dcap.com:8000/pls/OEFIN_portal30/porta
l30.home</apps_portal_url>
        <sysadminmail oa_var="s_sysadmin_mail"
osd="unix">oaoefin@dcap-dca-oraapp01.dcap.com</sysadminmail>
        <cookiedomain oa_var="s_cookie_domain">.dcap.com</cookiedomain>
        <topleveldomain_comment
oa_var="s_topleveldomain_comment">#</topleveldomain_comment>
        <iana_charset oa_var="s_iana_cset">UTF-8</iana_charset>
        <login_page
oa_var="s_login_page">http://wwwin-oefin.gslb.dcap.com:8000/oa_servlets/AppsLogin</login_p
age>
        <coredumpdest
oa_var="s_core_dest">/apps/local/OEFIN/iAS/Apache/Apache/logs</coredumpdest>
        <maxclients oa_var="s_maxclients">512</maxclients>
        <keepalive oa_var="s_keepalive">ON</keepalive>
        <keepalivetimeout oa_var="s_keepalive_timeout">15</keepalivetimeout>
        <maxrequestsperchild oa_var="s_maxrequests_perchild">0</maxrequestsperchild>
        <maxkeepaliverequests oa_var="s_maxkeepalive_requests">0</maxkeepaliverequests>
        <minspareservers oa_var="s_minspare_servers">5</minspareservers>
        <maxspareservers oa_var="s_maxspare_servers">10</maxspareservers>
        <maxsecurityconnections
oa_var="s_security_maxconnections">256</maxsecurityconnections>
        <forms_node_weight oa_var="s_forms_node_weight">1</forms_node_weight>
        <oacore_node_weight oa_var="s_oacore_node_weight">1</oacore_node_weight>
        <oacore_apjservmanual oa_var="s_apjservmanual">auto</oacore_apjservmanual>
        <oacore_nprocs oa_var="s_oacore_nprocs">1</oacore_nprocs>
        <servlet_nprocs oa_var="s_forms_servlet_nprocs">1</servlet_nprocs>
        <oacore_multiwebnode oa_var="s_multiwebnode">no</oacore_multiwebnode>
        <frmwrk_debug oa_var="s_fwk_debuglevel">0</frmwrk_debug>
        <csf_use_emap oa_var="s_csf_use_emap">N</csf_use_emap>
        <db_pon_maxpool oa_var="s_d_pon_max_pool_size">100</db_pon_maxpool>
        <db_pon_minpool oa_var="s_d_pon_min_pool_size">10</db_pon_minpool>
        <db_pool_growth_inc
oa_var="s_d_pon_pool_growth_increment">10</db_pool_growth_inc>
        <db_pool_growth_buf oa_var="s_d_pon_pool_growth_buffer">3</db_pool_growth_buf>
        <customsql_path
oa_var="s_customsql_path">/apps/oefin/appl_top/admin/OEFIN_dcap-dca-oraapp01/custom/sql</c
ustomsql_path>
        <cz_activemodel oa_var="s_cz_activemodel">/nolp|/nodp|/noatp|</cz_activemodel>
        <session_timeout oa_var="s_sesstimeout">1800000</session_timeout>
        <zone oa_var="s_emailcenter_comment">#</zone>
        <xmlsvcs_nprocs oa_var="s_xmlsvcs_nprocs">1</xmlsvcs_nprocs>
        <xmlsvcs_portrange oa_var="s_xmlsvcs_servlet_portrange"
oa_type="PORT">19000-19009</xmlsvcs_portrange>
```

```
         <xmlsvcs_node_weight oa_var="s_xmlsvcs_node_weight">1</xmlsvcs_node_weight>
         <servlet_autoreload oa_var="s_servlet_autoreload">false</servlet_autoreload>
         <java_showprogress oa_var="s_java_showprogress">false</java_showprogress>
         <funccompclass
oa_var="s_func_comp_loc">/apps/oefin/product/iAS/Apache/Jserv/servlets</funccompclass>
         <apjservvmtimeout oa_var="s_apjserv_vmtimeout">90</apjservvmtimeout>
         <oacorelog oa_var="s_oacorelog">false</oacorelog>
         <oadiscolog oa_var="s_oadiscolog">false</oadiscolog>
         <oaformslog oa_var="s_oaformslog">false</oaformslog>
         <oaxmllog oa_var="s_oaxmllog">false</oaxmllog>
         <servertokens oa_var="s_servertokens">Min</servertokens>
         <oemcomment oa_var="s_oemcomment">#</oemcomment>
         <oalogrotationtime oa_var="s_logrotationtime">86400</oalogrotationtime>
         <oxtainpoolsize oa_var="s_oxtainpool_size">1</oxtainpoolsize>
         <enableoxtaclientauth oa_var="s_enableoxtaclientauth">#</enableoxtaclientauth>
         <outboundthreads oa_var="s_outbound_threads">1</outboundthreads>
         <oacore_zone_name
oa_var="s_oacore_zone_name">rootdcap-dca-oraapp01</oacore_zone_name>
         <jserv_std_log oa_var="s_jserv_std_log">enable</jserv_std_log>
         <oafwkstartup oa_var="s_oafwkstartup">#</oafwkstartup>
         <apjservloglevel oa_var="s_apjservloglevel">warn</apjservloglevel>
         <sspdebuglevel oa_var="s_sspdebuglevel">5</sspdebuglevel>
         <sspdebugswitch oa_var="s_sspdebugswitch">OFF</sspdebugswitch>
         <oamname_space oa_var="s_oamname_space">oracle.apps.fnd.oam</oamname_space>
         <charset_jar oa_var="s_txk_wrapper_bin_charset_jar">charsets.jar</charset_jar>
         <jdbc_zip oa_var="s_txk_jdbc_zip">jdbc14.zip</jdbc_zip>
         <jms_cache oa_var="s_long_running_jvm">true</jms_cache>
         <xmlparser_name oa_var="s_xmlparserv2_name">xmlparserv2-904.zip</xmlparser_name>
         <jcache_port oa_var="s_java_object_cache_port" oa_type="PORT">12345</jcache_port>
         <fndcache_port_range oa_var="s_fnd_cache_port_range" oa_type="PORT"/>
         <jdbc_connections oa_var="s_fnd_max_jdbc_connections">500</jdbc_connections>
         <fnd_jdbc_stmt_cache_size
oa_var="s_fnd_jdbc_stmt_cache_size">200</fnd_jdbc_stmt_cache_size>
         <fnd_jdbc_stmt_cache_enable
oa_var="s_fnd_jdbc_stmt_cache_enable">TRUE</fnd_jdbc_stmt_cache_enable>
         <dbc_file_name oa_var="s_dbc_file_name">oefin</dbc_file_name>
         <jdbc_url
oa_var="s_apps_jdbc_connect_descriptor">jdbc:oracle:thin:@(DESCRIPTION=(LOAD_BALANCE=YES)(
FAILOVER=YES)(ADDRESS_LIST=(ADDRESS=(PROTOCOL=tcp)(HOST=db-oefin.gslb.dcap.com)(PORT=1521)
))(CONNECT_DATA=(SID=OEFIN)))</jdbc_url>
         <jdbc_connect_alias
oa_var="s_apps_jdbc_connect_alias">OEFIN_BALANCE</jdbc_connect_alias>
         <appserverid_authentication
oa_var="s_appserverid_authentication">SECURE</appserverid_authentication>
         <dbc_params>
            <fnd_jdbc_stmt_cache_free_mem oa_type="DBC"
oa_var="s_jdbc_cache_freemem">TRUE</fnd_jdbc_stmt_cache_free_mem>
            <fnd_jdbc_buffer_min oa_type="DBC"
oa_var="s_fnd_jdbc_buffermin">1</fnd_jdbc_buffer_min>
            <fnd_jdbc_buffer_max oa_type="DBC"
oa_var="s_fnd_jdbc_buffermax">5</fnd_jdbc_buffer_max>
            <fnd_jdbc_buffer_decay_interval oa_type="DBC"
oa_var="s_fnd_jdbc_buffer_decay_interval">300</fnd_jdbc_buffer_decay_interval>
            <fnd_jdbc_buffer_decay_size oa_type="DBC"
oa_var="s_fnd_jdbc_buffer_decay_size">5</fnd_jdbc_buffer_decay_size>
            <fnd_jdbc_usable_check oa_type="DBC"
oa_var="s_fnd_jdbc_usable_check">false</fnd_jdbc_usable_check>
            <fnd_jdbc_context_check oa_type="DBC"
oa_var="s_fnd_jdbc_context_check">true</fnd_jdbc_context_check>
            <fnd_jdbc_plsql_reset oa_type="DBC"
oa_var="s_fnd_jdbc_plsql_reset">false</fnd_jdbc_plsql_reset>
            <custom_dbc_param oa_var="s_custom_dbc_params"/>
         </dbc_params>
         <webentryurlprotocol oa_var="s_webentryurlprotocol">http</webentryurlprotocol>
```

```
         <apache_optionslink oa_var="s_options_symlinks">Options
-FollowSymLinks</apache_optionslink>
         <session_check_frequency
oa_var="s_sessionCheck_frequency">30000</session_check_frequency>
         <httploopback oa_var="s_httploopback">#</httploopback>
         <ohstimeout oa_var="s_ohstimeout">300</ohstimeout>
         <apache_loglevel oa_var="s_apache_loglevel">warn</apache_loglevel>
         <xmlparser_soap
oa_var="s_xmlparser_soap">/apps/oefin/common_top/java/xmlparserv2-904.zip</xmlparser_soap>
         <enablemodsecurity oa_var="s_enablemodsecurity"/>
         <servlet_init_timeout
oa_var="s_servlet_init_timeout">10000</servlet_init_timeout>
         <ohs_serveradmin oa_var="s_ohs_serveradmin">oaoefin@dcap.com</ohs_serveradmin>
         <httpclient_dontChunkRequests
oa_var="s_httpclient_dontChunkRequests">false</httpclient_dontChunkRequests>
      </oa_web_server>
      <oa_web_cache_server>
         <hostname oa_var="s_webcache_host">dcap-dca-oraapp01</hostname>
         <domain oa_var="s_webcache_domain">dcap.com</domain>
         <username oa_var="s_webcache_admin_userid">iasadmin</username>
         <url_protocol oa_var="s_webcache_url_protocol">http</url_protocol>
         <port oa_var="s_webcache_http_port" oa_type="PORT">8000</port>
         <port oa_var="s_webcache_https_port" oa_type="PORT">8000</port>
         <port oa_var="s_webcache_admin_port" oa_type="PORT">4000</port>
         <port oa_var="s_webcache_invalidation_port" oa_type="PORT">4001</port>
         <port oa_var="s_webcache_stats_port" oa_type="PORT">4002</port>
      </oa_web_cache_server>
   </oa_system>
   <oa_host>
      <host oa_var="s_hostname">dcap-dca-oraapp01</host>
      <domain oa_var="s_domainname">dcap.com</domain>
      <platform oa_var="s_platform" osd="Linux">Linux</platform>
      <pathsep oa_var="/" osd="unix">/</pathsep>
      <dbuser oa_var="s_dbuser" osd="unix">oaoefin</dbuser>
      <dbgroup oa_var="s_dbgroup" osd="unix">dba</dbgroup>
      <appsuser oa_var="s_appsuser" osd="unix">oaoefin</appsuser>
      <appsgroup oa_var="s_appsgroup" osd="unix">dba</appsgroup>
      <lib_ext oa_var="s_lib_ext" osd="Linux">so</lib_ext>
   </oa_host>
   <oa_install>
      <rapidwizloc
oa_var="s_rapidwizloc">/apps/11istage/CD/startCD/Disk1/rapidwiz</rapidwizloc>
      <installloc oa_var="s_installloc">/apps/11istage/CD/oraApps/Disk1</installloc>
      <clonestage oa_var="s_clonestage">/apps/11istage/CD/oraApps/Disk8</clonestage>
      <media oa_var="s_installedFrom">FS</media>
      <installthreads oa_var="s_nthreads">5</installthreads>
   </oa_install>
   <oa_environments>
      <adconfig>
         <adconfig_file
oa_var="s_adconfig_file">/apps/oefin/appl_top/admin/adconfig.txt</adconfig_file>
         <APPL_TOP oa_var="s_at_adconfig">/apps/oefin/appl_top</APPL_TOP>
         <APPL_TOP_CSET oa_var="s_at_cset">UTF8</APPL_TOP_CSET>
         <APPS_ENV_NAME oa_var="s_appsEnvName">OEFIN</APPS_ENV_NAME>
         <TIER_ADADMIN oa_var="s_isAdAdmin">YES</TIER_ADADMIN>
         <TIER_ADWEB oa_var="s_isAdWeb">YES</TIER_ADWEB>
         <TIER_ADFORMS oa_var="s_isAdForms">YES</TIER_ADFORMS>
         <TIER_ADNODE oa_var="s_isAdConc">YES</TIER_ADNODE>
         <APPL_TOP_NAME oa_var="s_atName">dcap</APPL_TOP_NAME>
         <TIER_ADFORMSDEV oa_var="s_isAdFormsDev">YES</TIER_ADFORMSDEV>
         <TIER_ADNODEDEV oa_var="s_isAdConcDev">YES</TIER_ADNODEDEV>
         <TIER_ADWEBDEV oa_var="s_isAdWebDev">YES</TIER_ADWEBDEV>
      </adconfig>
      <oa_environment type="generic_service">
```

```
         <DISPLAY oa_var="s_display" osd="unix">dcap-dca-oraapp01:0.0</DISPLAY>
         <java_awt_headless oa_var="s_java_awt_headless">false</java_awt_headless>
      </oa_environment>
      <oa_environment type="rapid_install">
         <HTML_TOP oa_var="s_html">/apps/oefin/common_top/html</HTML_TOP>
         <JDK_TOP
oa_var="s_jdktop">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04</JDK_TOP>
         <JRE_TOP
oa_var="s_jretop">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04</JRE_TOP>
         <PORTAL_TOP
oa_var="s_pt">/apps/oefin/common_top/portal/OEFIN_dcap-dca-oraapp01</PORTAL_TOP>
         <admin_dir oa_var="s_admindir">/apps/oefin/common_top/admin</admin_dir>
         <logs_dir
oa_var="s_logdir">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01</logs_dir>
         <out_dir
oa_var="s_outdir">/apps/oefin/common_top/admin/out/OEFIN_dcap-dca-oraapp01</out_dir>
         <temp_dir oa_var="s_temp">/apps/oefin/common_top/temp</temp_dir>
      </oa_environment>
      <oa_environment type="tools_home">
         <confighome oa_var="s_806config_home">/apps/local/OEFIN/8.0.6</confighome>
         <oa_env_file type="tools_home" oa_var="s_tools_home_file"
osd="unix">/apps/local/OEFIN/8.0.6/OEFIN_dcap-dca-oraapp01.env</oa_env_file>
         <ORACLE_HOME oa_var="s_tools_oh">/apps/oefin/product/8.0.6</ORACLE_HOME>
         <ADMIN_RESTRICTIONS oa_var="s_admin_restrictions">OFF</ADMIN_RESTRICTIONS>
         <LISTENER_PASSWORD oa_var="s_enable_listener_password">OFF</LISTENER_PASSWORD>
         <PATH oa_var="s_tools_path"
osd="Linux">/apps/oefin/product/8.0.6/bin:/usr/bin:/usr/sbin:/apps/oefin/common_top/util/j
ava/1.4/j2sdk1.4.2_04/bin:$PATH</PATH>
         <JAVA_HOME
oa_var="s_tools_java">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04</JAVA_HOME>
         <LD_LIBRARY_PATH oa_var="s_tools_ldlib"
osd="Linux">/apps/oefin/product/8.0.6/lib:/usr/X11R6/lib:/usr/openwin/lib</LD_LIBRARY_PATH
>
         <LINK_CNTRL oa_var="s_tools_linkctrl">L_PTHREADS_D7</LINK_CNTRL>
         <ORA_NLS oa_var="s_tools_oranls"
osd="unix">/apps/oefin/product/8.0.6/ocommon/nls/admin/data</ORA_NLS>
         <TNS_ADMIN oa_var="s_tools_tnsadmin"
osd="unix">/apps/local/OEFIN/8.0.6/network/admin/OEFIN_dcap-dca-oraapp01</TNS_ADMIN>
         <TWO_TASK oa_var="s_tools_twotask" osd="unix">OEFIN</TWO_TASK>
         <TOOLS_DB_TWOTASK
oa_var="s_tools_db_twotask">OEFIN_806_BALANCE</TOOLS_DB_TWOTASK>
         <CP_TWOTASK oa_var="s_cp_twotask">OEFIN</CP_TWOTASK>
         <IFILE
oa_var="s_ifile">/apps/local/OEFIN/8.0.6/network/admin/OEFIN_dcap-dca-oraapp01/OEFIN_dcap-
dca-oraapp01_ifile.ora</IFILE>
         <LISTENER_IFILE
oa_var="s_listener_ifile">/apps/local/OEFIN/8.0.6/network/admin/OEFIN_dcap-dca-oraapp01/OE
FIN_dcap-dca-oraapp01_listener_ifile.ora</LISTENER_IFILE>
         <ORACLE_LOCALPREFERENCE
oa_var="s_tools_locpref">/apps/oefin/product/8.0.6/tools/admin</ORACLE_LOCALPREFERENCE>
         <NLS_LANG oa_var="s_tools_nlslang">American_America.UTF8</NLS_LANG>
         <SQLNET_EXPIRE_TIME oa_var="s_sqlnet_expire_time"
osd="UNIX">0</SQLNET_EXPIRE_TIME>
         <FORMS60_WALLET
oa_var="s_forms60_wallet">/apps/local/OEFIN/8.0.6/forms60/wallet</FORMS60_WALLET>
         <FORMS60_HTTPS_NEGOTIATE_DOWN
oa_var="s_forms60_https_negotiate_down">TRUE</FORMS60_HTTPS_NEGOTIATE_DOWN>
      </oa_environment>
      <oa_environment type="web_home">
         <confighome oa_var="s_iASconfig_home">/apps/local/OEFIN/iAS</confighome>
         <oa_env_file type="web_home" oa_var="s_web_home_file"
osd="unix">/apps/local/OEFIN/iAS/OEFIN_dcap-dca-oraapp01.env</oa_env_file>
         <ORACLE_HOME oa_var="s_weboh_oh">/apps/oefin/product/iAS</ORACLE_HOME>
         <APACHE_TOP oa_var="s_tp">/apps/oefin/product/iAS</APACHE_TOP>
```

```
         <PATH oa_var="s_weboh_path"
osd="Linux">/apps/oefin/product/iAS/bin:/usr/bin:/usr/sbin:/apps/oefin/product/iAS/Apache/
jdk/bin:$PATH</PATH>
         <JAVA_HOME oa_var="s_weboh_java">/apps/oefin/product/iAS/Apache/jdk</JAVA_HOME>
         <LD_LIBRARY_PATH oa_var="s_weboh_ldlib"
osd="Linux">/apps/oefin/product/iAS/lib:/usr/X11R6/lib</LD_LIBRARY_PATH>
         <LINK_CNTRL oa_var="s_weboh_linkctrl">L_PTHREADS_D7</LINK_CNTRL>
         <ORA_NLS oa_var="s_weboh_oranls"
osd="unix">/apps/oefin/product/iAS/ocommon/nls/admin/data</ORA_NLS>
         <TNS_ADMIN
oa_var="s_weboh_tnsadmin">/apps/local/OEFIN/iAS/network/admin/OEFIN_dcap-dca-oraapp01</TNS
_ADMIN>
         <TWO_TASK oa_var="s_weboh_twotask" osd="unix">OEFIN</TWO_TASK>
         <WEBOH_DB_TWOTASK
oa_var="s_weboh_db_twotask">OEFIN_806_BALANCE</WEBOH_DB_TWOTASK>
         <WEB_IFILE
oa_var="s_ias_ifile">/apps/local/OEFIN/iAS/network/admin/OEFIN_dcap-dca-oraapp01/OEFIN_dca
p-dca-oraapp01_ifile.ora</WEB_IFILE>
         <ORACLE_LOCALPREFERENCE
oa_var="s_weboh_locpref">/apps/oefin/product/iAS/tools/admin</ORACLE_LOCALPREFERENCE>
         <NLS_LANG oa_var="s_weboh_nlslang">American_America.UTF8</NLS_LANG>
         <WV_GATEWAY_CFG
oa_var="s_wv_gateway_cfg">/apps/local/OEFIN/iAS/Apache/modplsql/cfg/wdbsvr.app</WV_GATEWAY
_CFG>
         <FNDFS_PARAMS oa_var="s_fndfs_params"/>
      </oa_environment>
      <oa_environment type="adovars">
         <oa_env_file type="adovars" oa_var="s_adovars_file"
osd="unix">/apps/oefin/appl_top/admin/adovars.env</oa_env_file>
         <JAVA_TOP oa_var="s_javatop">/apps/oefin/common_top/java</JAVA_TOP>
         <OA_JDK_TOP
oa_var="s_oajdktop">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04</OA_JDK_TOP>
         <OA_JRE_TOP
oa_var="s_oajretop">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04</OA_JRE_TOP>
         <AF_JRE_TOP
oa_var="s_afjretop">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04</AF_JRE_TOP>
         <CLASSPATH oa_var="s_adovar_classpath"
osd="unix">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/lib/dt.jar:/apps/oefin/commo
n_top/util/java/1.4/j2sdk1.4.2_04/lib/tools.jar:/apps/oefin/common_top/util/java/1.4/j2sdk
1.4.2_04/jre/lib/rt.jar:/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/jre/lib/charset
s.jar:/apps/oefin/common_top/java/appsborg2.zip:/apps/oefin/common_top/java/apps.zip:/apps
/oefin/product/8.0.6/forms60/java:/apps/oefin/common_top/java</CLASSPATH>
         <AF_CLASSPATH oa_var="s_adovar_afclasspath"
osd="unix">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/lib/dt.jar:/apps/oefin/commo
n_top/util/java/1.4/j2sdk1.4.2_04/lib/tools.jar:/apps/oefin/common_top/util/java/1.4/j2sdk
1.4.2_04/jre/lib/rt.jar:/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/jre/lib/charset
s.jar:/apps/oefin/common_top/java/appsborg2.zip:/apps/oefin/common_top/java/apps.zip:/apps
/oefin/product/8.0.6/forms60/java:/apps/oefin/common_top/java</AF_CLASSPATH>
         <LD_LIBRARY_PATH oa_var="s_adovar_ldlib"
osd="Linux">/apps/oefin/product/8.0.6/network/jre11/lib/i686/native_threads:/apps/oefin/pr
oduct/8.0.6/network/jre11/lib/linux/native_threads:/apps/oefin/appl_top/cz/11.5.0/bin:${LD
_LIBRARY_PATH:=}</LD_LIBRARY_PATH>
         <OAH_TOP oa_var="s_oahtop">/apps/oefin/common_top</OAH_TOP>
         <OAD_TOP oa_var="s_oadtop">/apps/oefin/common_top</OAD_TOP>
         <OAM_TOP oa_var="s_oamtop">/apps/oefin/common_top/java/oracle/apps</OAM_TOP>
         <ADJVAPRG oa_var="s_adjvaprg"
osd="UNIX">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/bin/java</ADJVAPRG>
         <AFJVAPRG oa_var="s_afjvaprg"
osd="UNIX">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/bin/java</AFJVAPRG>
         <ADJREOPTS oa_var="s_adjreopts">-ms128m -mx256m</ADJREOPTS>
         <ADJRIOPTS oa_var="s_adjriopts">-mx512m</ADJRIOPTS>
         <APPSJREOPTS oa_var="s_appsjreopts"/>
         <OA_HTML oa_var="s_oahtml">/apps/oefin/common_top/html</OA_HTML>
         <OA_SECURE oa_var="s_oasecure">/apps/oefin/common_top/secure</OA_SECURE>
```

```
        <OA_MEDIA
oa_var="s_oamedia">/apps/oefin/common_top/java/oracle/apps/media</OA_MEDIA>
        <OA_DOC oa_var="s_oadoc">/apps/oefin/common_top/doc</OA_DOC>
        <OA_JAVA oa_var="s_oajava">/apps/oefin/common_top/java</OA_JAVA>
        <oa_htmlbin oa_var="s_oahtmlbin">/apps/oefin/common_top/html/bin</oa_htmlbin>
        <AFJCPARG oa_var="s_afjcparg"/>
        <AFJSMARG oa_var="s_afjsmarg"/>
    </oa_environment>
    <oa_environment type="applsys">
        <oa_env_file type="appsora" oa_var="s_appsora_file"
osd="unix">/apps/oefin/appl_top/APPSOEFIN_dcap-dca-oraapp01.env</oa_env_file>
        <oa_env_file type="applsys" oa_var="s_applsys_file"
osd="unix">/apps/oefin/appl_top/OEFIN_dcap-dca-oraapp01.env</oa_env_file>
        <oa_env_file type="custom" oa_var="s_custom_file"
osd="unix">/apps/oefin/appl_top/customOEFIN_dcap-dca-oraapp01.env</oa_env_file>
        <APPLFENV oa_var="s_applfenv">OEFIN_dcap-dca-oraapp01.env</APPLFENV>
        <APPLFSTT
oa_var="s_applfstt">OEFIN_FO;OEFIN_806_BALANCE;OEFIN;OEFIN_BALANCE</APPLFSTT>
        <PLATFORM oa_var="s_oraPlatform" osd="Linux">LINUX</PLATFORM>
        <PLATFORM_STUB oa_var="s_platStub" osd="Linux">linux</PLATFORM_STUB>
        <APPL_TOP oa_var="s_at">/apps/oefin/appl_top</APPL_TOP>
        <APPL_TOP2 oa_var="s_at2">/apps/oefin/appl_top</APPL_TOP2>
        <APPL_TOP3 oa_var="s_at3">/apps/oefin/appl_top</APPL_TOP3>
        <APPL_TOP4 oa_var="s_at4">/apps/oefin/appl_top</APPL_TOP4>
        <ADMIN_SID oa_var="s_admin_sid">/apps/oefin/appl_top/admin/OEFIN</ADMIN_SID>
        <FNDNAM oa_var="s_fndnam">APPS</FNDNAM>
        <APPCPNAM oa_var="s_appcpnam">REQID</APPCPNAM>
        <APPLMAIL oa_var="s_applmail">NONE</APPLMAIL>
        <APPLDCP oa_var="s_appldcp">OFF</APPLDCP>
        <APPLCSF oa_var="s_applcsf">/apps/oefin/common_top/admin</APPLCSF>
        <APPLLOG oa_var="s_appllog">log/OEFIN_dcap-dca-oraapp01</APPLLOG>
        <APPLOUT oa_var="s_applout">out/OEFIN_dcap-dca-oraapp01</APPLOUT>
        <APPLRGF
oa_var="s_applrgf">/apps/oefin/common_top/rgf/OEFIN_dcap-dca-oraapp01</APPLRGF>
        <APPLTMP oa_var="s_appltmp">/apps/oefin/common_top/temp</APPLTMP>
        <APPLPTMP oa_var="s_applptmp" osd="UNIX">/usr/tmp</APPLPTMP>
        <NLS_LANG oa_var="s_nlslang">American_America.UTF8</NLS_LANG>
        <NLS_DATE_FORMAT oa_var="s_nlsdate">DD-MON-RR</NLS_DATE_FORMAT>
        <NLS_NUMERIC_CHARACTERS oa_var="s_oa_nlsnumchar">.,</NLS_NUMERIC_CHARACTERS>
        <NLS_SORT oa_var="s_nlssort">BINARY</NLS_SORT>
        <FORMS60_OVERRIDE_ENV
oa_var="s_f60override_env">NLS_LANG,NLS_NUMERIC_CHARACTERS,NLS_SORT,NLS_DATE_LANGUAGE,NLS_
DATE_FORMAT,FORMS60_USER_DATE_FORMAT,FORMS60_USER_DATETIME_FORMAT,FORMS60_OUTPUT_DATE_FORM
AT,FORMS60_OUTPUT_DATETIME_FORMAT,FORMS60_ERROR_DATE_FORMAT,FORMS60_ERROR_DATETIME_FORMAT,
FORMS60_TZFILE,FORMS60_DATETIME_SERVER_TZ,FORMS60_DATETIME_LOCAL_TZ,FORMS60_USER_CALENDAR<
/FORMS60_OVERRIDE_ENV>
        <FORMS60_MODULE_PATH
oa_var="s_f60module_path">/apps/oefin/appl_top/fnd/11.5.0/forms</FORMS60_MODULE_PATH>
        <REPORTS60_TMP oa_var="s_reptmp">/apps/oefin/common_top/temp</REPORTS60_TMP>
        <REPORTS60_PATH oa_var="s_reppath"
osd="UNIX">/apps/oefin/appl_top/au/11.5.0/plsql:/apps/oefin/appl_top/fnd/11.5.0/reports:/a
pps/oefin/appl_top/au/11.5.0/reports:/apps/oefin/appl_top/au/11.5.0/graphs:/apps/oefin/pro
duct/8.0.6/reports60/admin/printer</REPORTS60_PATH>
        <GRAPHICS60_PATH
oa_var="s_gr60path">/apps/oefin/appl_top/au/11.5.0/graphs</GRAPHICS60_PATH>
        <ORAPLSQLLOADPATH
oa_var="s_plsqlldpath">/apps/oefin/appl_top/au/11.5.0/graphs</ORAPLSQLLOADPATH>
        <FORMS60_MAPPING
oa_var="s_f60map">http://dcap-dca-oraapp01.dcap.com:8000/OA_TEMP</FORMS60_MAPPING>
        <CNTL_BREAK oa_var="s_ctrlbrk">ON</CNTL_BREAK>
        <FORMS60_MESSAGE_ENCRYPTION oa_var="s_f60encr">TRUE</FORMS60_MESSAGE_ENCRYPTION>
        <FORMS60_OUTPUT oa_var="s_f60out">/apps/oefin/common_top/temp</FORMS60_OUTPUT>
        <FORMS60_SESSION oa_var="s_f60ses">TRUE</FORMS60_SESSION>
        <FORMS60_OAM_FRD oa_var="s_f60_oam_frd">OFF</FORMS60_OAM_FRD>
```

```
        <FORMS60_RTI_DIR
oa_var="s_formsrtidir">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01</FORMS60_R
TI_DIR>
        <ORACLE_TERM oa_var="s_oraterm">vt220</ORACLE_TERM>
        <FORMS60_APPSLIBS oa_var="s_f60appslib">APPCORE FNDSQF APPDAYPK APPFLDR GLCORE
HR_GEN HR_SPEC ARXCOVER</FORMS60_APPSLIBS>
        <FORMS60_FORCE_MENU_MNEMONICS oa_var="s_f60fmm">0</FORMS60_FORCE_MENU_MNEMONICS>
        <FORMS60_PATH oa_var="s_f60path"
osd="unix">/apps/oefin/appl_top/au/11.5.0/resource:/apps/oefin/appl_top/au/11.5.0/resource
/stub</FORMS60_PATH>
        <FORMS60_TIMEOUT oa_var="s_f60time">5</FORMS60_TIMEOUT>
        <FORMS60_USER_DATE_FORMAT
oa_var="s_f60udate">MM/DD/RRRR</FORMS60_USER_DATE_FORMAT>
        <FORMS60_USER_DATETIME_FORMAT oa_var="s_f60udatetime">MM/DD/RRRR
HH24:MI:SS</FORMS60_USER_DATETIME_FORMAT>
        <APPL_CPLEX_LICDIR
oa_var="s_cplexlic">/apps/oefin/appl_top/admin/cplex</APPL_CPLEX_LICDIR>
        <FORMS60_WEB_CONFIG_FILE
oa_var="s_f60webcfg">/apps/oefin/common_top/html/bin/appsweb_OEFIN_dcap-dca-oraapp01.cfg</
FORMS60_WEB_CONFIG_FILE>
        <FORMS60_BLOCKING_LONGLIST
oa_var="s_f60blocklist">FALSE</FORMS60_BLOCKING_LONGLIST>
        <FORMS60_LOV_INITIAL oa_var="s_f60lov">5000</FORMS60_LOV_INITIAL>
        <FORMS60_CATCHTERM oa_var="s_f60catchterm">1</FORMS60_CATCHTERM>
        <FORMS60_DISABLE_UNPAD_LOV
oa_var="s_forms60_disable_unpad_lov">FALSE</FORMS60_DISABLE_UNPAD_LOV>
        <FORMS60_LOV_MINIMUM oa_var="s_f60lovminimum">1000</FORMS60_LOV_MINIMUM>
        <FORMS60_LOV_WEIGHT oa_var="s_f60lovweight">16</FORMS60_LOV_WEIGHT>
        <FORMS60_NONBLOCKING_SLEEP
oa_var="s_f60nonblockingsleep">100</FORMS60_NONBLOCKING_SLEEP>
        <FORMS60_BLOCK_URL_CHARACTERS
oa_var="s_f60blockurlchar">%0a,%0d,!,%21,",%22,(,%28,),%29,;,[,%5b,],%5d,{,%7b,|,%7c,},%7d
,%7f,>,%3c,&lt;,%3e</FORMS60_BLOCK_URL_CHARACTERS>
        <fnd_secure
oa_var="s_fnd_secure">/apps/oefin/appl_top/fnd/11.5.0/secure/OEFIN_dcap-dca-oraapp01</fnd_
secure>
        <AD_TOP oa_var="s_adtop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/ad/11.5.0</AD_TOP>
        <AK_TOP oa_var="s_aktop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ak/11.5.0</AK_TOP>
        <AU_TOP oa_var="s_autop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/au/11.5.0</AU_TOP>
        <FND_TOP oa_var="s_fndtop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/fnd/11.5.0</FND_TOP>
        <ABM_TOP oa_var="s_abmtop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/abm/11.5.0</ABM_TOP>
        <AHL_TOP oa_var="s_ahltop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/ahl/11.5.0</AHL_TOP>
        <AHM_TOP oa_var="s_ahmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ahm/11.5.0</AHM_TOP>
        <ALR_TOP oa_var="s_alrtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/alr/11.5.0</ALR_TOP>
        <AMF_TOP oa_var="s_amftop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/amf/11.5.0</AMF_TOP>
        <AMS_TOP oa_var="s_amstop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/ams/11.5.0</AMS_TOP>
        <AMV_TOP oa_var="s_amvtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/amv/11.5.0</AMV_TOP>
        <AMW_TOP oa_var="s_amwtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/amw/11.5.0</AMW_TOP>
        <AP_TOP oa_var="s_aptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ap/11.5.0</AP_TOP>
        <AR_TOP oa_var="s_artop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/ar/11.5.0</AR_TOP>
```

```
        <AS_TOP oa_var="s_astop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/as/11.5.0</AS_TOP>
        <ASN_TOP oa_var="s_asntop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/asn/11.5.0</ASN_TOP>
        <ASF_TOP oa_var="s_asftop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/asf/11.5.0</ASF_TOP>
        <ASG_TOP oa_var="s_asgtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/asg/11.5.0</ASG_TOP>
        <ASL_TOP oa_var="s_asltop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/asl/11.5.0</ASL_TOP>
        <ASO_TOP oa_var="s_asotop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/aso/11.5.0</ASO_TOP>
        <ASP_TOP oa_var="s_asptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/asp/11.5.0</ASP_TOP>
        <AST_TOP oa_var="s_asttop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ast/11.5.0</AST_TOP>
        <AX_TOP oa_var="s_axtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ax/11.5.0</AX_TOP>
        <AZ_TOP oa_var="s_aztop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/az/11.5.0</AZ_TOP>
        <BEN_TOP oa_var="s_bentop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ben/11.5.0</BEN_TOP>
        <BIC_TOP oa_var="s_bictop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/bic/11.5.0</BIC_TOP>
        <BIL_TOP oa_var="s_biltop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/bil/11.5.0</BIL_TOP>
        <BIM_TOP oa_var="s_bimtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/bim/11.5.0</BIM_TOP>
        <BIS_TOP oa_var="s_bistop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/bis/11.5.0</BIS_TOP>
        <BIV_TOP oa_var="s_bivtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/biv/11.5.0</BIV_TOP>
        <BIX_TOP oa_var="s_bixtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/bix/11.5.0</BIX_TOP>
        <BNE_TOP oa_var="s_bnetop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/bne/11.5.0</BNE_TOP>
        <BOM_TOP oa_var="s_bomtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/bom/11.5.0</BOM_TOP>
        <BSC_TOP oa_var="s_bsctop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/bsc/11.5.0</BSC_TOP>
        <CCT_TOP oa_var="s_ccttop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/cct/11.5.0</CCT_TOP>
        <CE_TOP oa_var="s_cetop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ce/11.5.0</CE_TOP>
        <CHV_TOP oa_var="s_chvtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/chv/11.5.0</CHV_TOP>
        <CLN_TOP oa_var="s_clntop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/cln/11.5.0</CLN_TOP>
        <CN_TOP oa_var="s_cntop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/cn/11.5.0</CN_TOP>
        <CRP_TOP oa_var="s_crptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/crp/11.5.0</CRP_TOP>
        <CS_TOP oa_var="s_cstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cs/11.5.0</CS_TOP>
        <CSC_TOP oa_var="s_csctop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/csc/11.5.0</CSC_TOP>
        <CSD_TOP oa_var="s_csdtop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/csd/11.5.0</CSD_TOP>
        <CSE_TOP oa_var="s_csetop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/cse/11.5.0</CSE_TOP>
        <CSF_TOP oa_var="s_csftop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/csf/11.5.0</CSF_TOP>
        <CSI_TOP oa_var="s_csitop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/csi/11.5.0</CSI_TOP>
```

```
        <CSL_TOP oa_var="s_csltop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/csl/11.5.0</CSL_TOP>
        <CSM_TOP oa_var="s_csmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/csm/11.5.0</CSM_TOP>
        <CSP_TOP oa_var="s_csptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/csp/11.5.0</CSP_TOP>
        <CSR_TOP oa_var="s_csrtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/csr/11.5.0</CSR_TOP>
        <CSS_TOP oa_var="s_csstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/css/11.5.0</CSS_TOP>
        <CUA_TOP oa_var="s_cuatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cua/11.5.0</CUA_TOP>
        <CUE_TOP oa_var="s_cuetop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cue/11.5.0</CUE_TOP>
        <CUF_TOP oa_var="s_cuftop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cuf/11.5.0</CUF_TOP>
        <CUG_TOP oa_var="s_cugtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cug/11.5.0</CUG_TOP>
        <CUI_TOP oa_var="s_cuitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cui/11.5.0</CUI_TOP>
        <CUN_TOP oa_var="s_cuntop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cun/11.5.0</CUN_TOP>
        <CUP_TOP oa_var="s_cuptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cup/11.5.0</CUP_TOP>
        <CUS_TOP oa_var="s_custop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cus/11.5.0</CUS_TOP>
        <CZ_TOP oa_var="s_cztop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/cz/11.5.0</CZ_TOP>
        <DDD_TOP oa_var="s_dddtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ddd/11.5.0</DDD_TOP>
        <DOM_TOP oa_var="s_domtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/dom/11.5.0</DOM_TOP>
        <DT_TOP oa_var="s_dttop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/dt/11.5.0</DT_TOP>
        <EAA_TOP oa_var="s_eaatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/eaa/11.5.0</EAA_TOP>
        <EAM_TOP oa_var="s_eamtop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/eam/11.5.0</EAM_TOP>
        <EC_TOP oa_var="s_ectop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ec/11.5.0</EC_TOP>
        <ECX_TOP oa_var="s_ecxtop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/ecx/11.5.0</ECX_TOP>
        <EDR_TOP oa_var="s_edrtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/edr/11.5.0</EDR_TOP>
        <EGO_TOP oa_var="s_egotop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ego/11.5.0</EGO_TOP>
        <ENG_TOP oa_var="s_engtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/eng/11.5.0</ENG_TOP>
        <ENI_TOP oa_var="s_enitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/eni/11.5.0</ENI_TOP>
        <EVM_TOP oa_var="s_evmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/evm/11.5.0</EVM_TOP>
        <FA_TOP oa_var="s_fatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/fa/11.5.0</FA_TOP>
        <FEM_TOP oa_var="s_femtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/fem/11.5.0</FEM_TOP>
        <FF_TOP oa_var="s_fftop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ff/11.5.0</FF_TOP>
        <FII_TOP oa_var="s_fiitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/fii/11.5.0</FII_TOP>
        <FLM_TOP oa_var="s_flmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/flm/11.5.0</FLM_TOP>
        <FPA_TOP oa_var="s_fpatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/fpa/11.5.0</FPA_TOP>
```

```
        <FPT_TOP oa_var="s_fpttop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/fpt/11.5.0</FPT_TOP>
        <FRM_TOP oa_var="s_frmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/frm/11.5.0</FRM_TOP>
        <FTE_TOP oa_var="s_ftetop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/fte/11.5.0</FTE_TOP>
        <FUN_TOP oa_var="s_funtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/fun/11.5.0</FUN_TOP>
        <FV_TOP oa_var="s_fvtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/fv/11.5.0</FV_TOP>
        <GCS_TOP oa_var="s_gcstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gcs/11.5.0</GCS_TOP>
        <GHR_TOP oa_var="s_ghrtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ghr/11.5.0</GHR_TOP>
        <GL_TOP oa_var="s_gltop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/gl/11.5.0</GL_TOP>
        <GMA_TOP oa_var="s_gmatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gma/11.5.0</GMA_TOP>
        <GMD_TOP oa_var="s_gmdtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gmd/11.5.0</GMD_TOP>
        <GME_TOP oa_var="s_gmetop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gme/11.5.0</GME_TOP>
        <GMF_TOP oa_var="s_gmftop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gmf/11.5.0</GMF_TOP>
        <GMI_TOP oa_var="s_gmitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gmi/11.5.0</GMI_TOP>
        <GML_TOP oa_var="s_gmltop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gml/11.5.0</GML_TOP>
        <GMP_TOP oa_var="s_gmptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gmp/11.5.0</GMP_TOP>
        <GMS_TOP oa_var="s_gmstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gms/11.5.0</GMS_TOP>
        <GR_TOP oa_var="s_grtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/gr/11.5.0</GR_TOP>
        <HRI_TOP oa_var="s_hritop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/hri/11.5.0</HRI_TOP>
        <HXC_TOP oa_var="s_hxctop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/hxc/11.5.0</HXC_TOP>
        <HXT_TOP oa_var="s_hxttop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/hxt/11.5.0</HXT_TOP>
        <IA_TOP oa_var="s_iatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ia/11.5.0</IA_TOP>
        <IBA_TOP oa_var="s_ibatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/iba/11.5.0</IBA_TOP>
        <IBC_TOP oa_var="s_ibctop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ibc/11.5.0</IBC_TOP>
        <IBE_TOP oa_var="s_ibetop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ibe/11.5.0</IBE_TOP>
        <IBP_TOP oa_var="s_ibptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ibp/11.5.0</IBP_TOP>
        <IBU_TOP oa_var="s_ibutop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ibu/11.5.0</IBU_TOP>
        <IBY_TOP oa_var="s_ibytop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/iby/11.5.0</IBY_TOP>
        <ICX_TOP oa_var="s_icxtop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/icx/11.5.0</ICX_TOP>
        <IEB_TOP oa_var="s_iebtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ieb/11.5.0</IEB_TOP>
        <IEC_TOP oa_var="s_iectop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/iec/11.5.0</IEC_TOP>
        <IEM_TOP oa_var="s_iemtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/iem/11.5.0</IEM_TOP>
        <IEO_TOP oa_var="s_ieotop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/ieo/11.5.0</IEO_TOP>
```

```
        <IES_TOP oa_var="s_iestop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/ies/11.5.0</IES_TOP>
        <IEU_TOP oa_var="s_ieutop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ieu/11.5.0</IEU_TOP>
        <IEX_TOP oa_var="s_iextop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/iex/11.5.0</IEX_TOP>
        <IGC_TOP oa_var="s_igctop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/igc/11.5.0</IGC_TOP>
        <IGF_TOP oa_var="s_igftop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/igf/11.5.0</IGF_TOP>
        <IGI_TOP oa_var="s_igitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/igi/11.5.0</IGI_TOP>
        <IGS_TOP oa_var="s_igstop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/igs/11.5.0</IGS_TOP>
        <IGW_TOP oa_var="s_igwtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/igw/11.5.0</IGW_TOP>
        <IMC_TOP oa_var="s_imctop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/imc/11.5.0</IMC_TOP>
        <IMT_TOP oa_var="s_imttop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/imt/11.5.0</IMT_TOP>
        <INV_TOP oa_var="s_invtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/inv/11.5.0</INV_TOP>
        <IPA_TOP oa_var="s_ipatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ipa/11.5.0</IPA_TOP>
        <IPD_TOP oa_var="s_ipdtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ipd/11.5.0</IPD_TOP>
        <ISC_TOP oa_var="s_isctop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/isc/11.5.0</ISC_TOP>
        <ITG_TOP oa_var="s_itgtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/itg/11.5.0</ITG_TOP>
        <JA_TOP oa_var="s_jatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ja/11.5.0</JA_TOP>
        <JE_TOP oa_var="s_jetop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/je/11.5.0</JE_TOP>
        <JG_TOP oa_var="s_jgtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/jg/11.5.0</JG_TOP>
        <JL_TOP oa_var="s_jltop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/jl/11.5.0</JL_TOP>
        <JTF_TOP oa_var="s_jtftop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/jtf/11.5.0</JTF_TOP>
        <JTM_TOP oa_var="s_jtmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/jtm/11.5.0</JTM_TOP>
        <JTS_TOP oa_var="s_jtstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/jts/11.5.0</JTS_TOP>
        <LNS_TOP oa_var="s_lnstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/lns/11.5.0</LNS_TOP>
        <ME_TOP oa_var="s_metop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/me/11.5.0</ME_TOP>
        <MFG_TOP oa_var="s_mfgtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/mfg/11.5.0</MFG_TOP>
        <MRP_TOP oa_var="s_mrptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/mrp/11.5.0</MRP_TOP>
        <MSC_TOP oa_var="s_msctop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/msc/11.5.0</MSC_TOP>
        <MSD_TOP oa_var="s_msdtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/msd/11.5.0</MSD_TOP>
        <MSO_TOP oa_var="s_msotop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/mso/11.5.0</MSO_TOP>
        <MSR_TOP oa_var="s_msrtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/msr/11.5.0</MSR_TOP>
        <MST_TOP oa_var="s_msttop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/mst/11.5.0</MST_TOP>
        <MWA_TOP oa_var="s_mwatop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/mwa/11.5.0</MWA_TOP>
```

```
        <OE_TOP oa_var="s_oetop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/oe/11.5.0</OE_TOP>
        <OKB_TOP oa_var="s_okbtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/okb/11.5.0</OKB_TOP>
        <OKC_TOP oa_var="s_okctop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/okc/11.5.0</OKC_TOP>
        <OKE_TOP oa_var="s_oketop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/oke/11.5.0</OKE_TOP>
        <OKI_TOP oa_var="s_okitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/oki/11.5.0</OKI_TOP>
        <OKL_TOP oa_var="s_okltop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/okl/11.5.0</OKL_TOP>
        <OKO_TOP oa_var="s_okotop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/oko/11.5.0</OKO_TOP>
        <OKR_TOP oa_var="s_okrtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/okr/11.5.0</OKR_TOP>
        <OKS_TOP oa_var="s_okstop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/oks/11.5.0</OKS_TOP>
        <OKX_TOP oa_var="s_okxtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/okx/11.5.0</OKX_TOP>
        <ONT_TOP oa_var="s_onttop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/ont/11.5.0</ONT_TOP>
        <OPI_TOP oa_var="s_opitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/opi/11.5.0</OPI_TOP>
        <OTA_TOP oa_var="s_otatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ota/11.5.0</OTA_TOP>
        <OZF_TOP oa_var="s_ozftop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/ozf/11.5.0</OZF_TOP>
        <OZP_TOP oa_var="s_ozptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ozp/11.5.0</OZP_TOP>
        <OZS_TOP oa_var="s_ozstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ozs/11.5.0</OZS_TOP>
        <PA_TOP oa_var="s_patop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/pa/11.5.0</PA_TOP>
        <PAY_TOP oa_var="s_paytop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pay/11.5.0</PAY_TOP>
        <PER_TOP oa_var="s_pertop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/per/11.5.0</PER_TOP>
        <PJI_TOP oa_var="s_pjitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pji/11.5.0</PJI_TOP>
        <PJM_TOP oa_var="s_pjmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pjm/11.5.0</PJM_TOP>
        <PMI_TOP oa_var="s_pmitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pmi/11.5.0</PMI_TOP>
        <PN_TOP oa_var="s_pntop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pn/11.5.0</PN_TOP>
        <PO_TOP oa_var="s_potop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/po/11.5.0</PO_TOP>
        <POA_TOP oa_var="s_poatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/poa/11.5.0</POA_TOP>
        <POM_TOP oa_var="s_pomtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pom/11.5.0</POM_TOP>
        <PON_TOP oa_var="s_pontop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pon/11.5.0</PON_TOP>
        <POS_TOP oa_var="s_postop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pos/11.5.0</POS_TOP>
        <PQH_TOP oa_var="s_pqhtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pqh/11.5.0</PQH_TOP>
        <PQP_TOP oa_var="s_pqptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pqp/11.5.0</PQP_TOP>
        <PRP_TOP oa_var="s_prptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/prp/11.5.0</PRP_TOP>
        <PSA_TOP oa_var="s_psatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/psa/11.5.0</PSA_TOP>
```

```
        <PSB_TOP oa_var="s_psbtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/psb/11.5.0</PSB_TOP>
        <PSP_TOP oa_var="s_psptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/psp/11.5.0</PSP_TOP>
        <PV_TOP oa_var="s_pvtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/pv/11.5.0</PV_TOP>
        <QA_TOP oa_var="s_qatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/qa/11.5.0</QA_TOP>
        <QOT_TOP oa_var="s_qottop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/qot/11.5.0</QOT_TOP>
        <QP_TOP oa_var="s_qptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/qp/11.5.0</QP_TOP>
        <QRM_TOP oa_var="s_qrmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/qrm/11.5.0</QRM_TOP>
        <RCM_TOP oa_var="s_rcmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/rcm/11.5.0</RCM_TOP>
        <RG_TOP oa_var="s_rgtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/rg/11.5.0</RG_TOP>
        <RHX_TOP oa_var="s_rhxtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/rhx/11.5.0</RHX_TOP>
        <RLA_TOP oa_var="s_rlatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/rla/11.5.0</RLA_TOP>
        <RLM_TOP oa_var="s_rlmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/rlm/11.5.0</RLM_TOP>
        <SHT_TOP oa_var="s_shttop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/sht/11.5.0</SHT_TOP>
        <SSP_TOP oa_var="s_ssptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/ssp/11.5.0</SSP_TOP>
        <VEA_TOP oa_var="s_veatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/vea/11.5.0</VEA_TOP>
        <VEH_TOP oa_var="s_vehtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/veh/11.5.0</VEH_TOP>
        <WIP_TOP oa_var="s_wiptop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/wip/11.5.0</WIP_TOP>
        <WMS_TOP oa_var="s_wmstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/wms/11.5.0</WMS_TOP>
        <WPS_TOP oa_var="s_wpstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/wps/11.5.0</WPS_TOP>
        <WSH_TOP oa_var="s_wshtop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/wsh/11.5.0</WSH_TOP>
        <WSM_TOP oa_var="s_wsmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/wsm/11.5.0</WSM_TOP>
        <XDP_TOP oa_var="s_xdptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xdp/11.5.0</XDP_TOP>
        <XDO_TOP oa_var="s_xdotop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xdo/11.5.0</XDO_TOP>
        <XLA_TOP oa_var="s_xlatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xla/11.5.0</XLA_TOP>
        <XLE_TOP oa_var="s_xletop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xle/11.5.0</XLE_TOP>
        <XNB_TOP oa_var="s_xnbtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xnb/11.5.0</XNB_TOP>
        <XNC_TOP oa_var="s_xnctop" oa_type="PROD_TOP"
oa_enabled="TRUE">/apps/oefin/appl_top/xnc/11.5.0</XNC_TOP>
        <XNI_TOP oa_var="s_xnitop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xni/11.5.0</XNI_TOP>
        <XNM_TOP oa_var="s_xnmtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xnm/11.5.0</XNM_TOP>
        <XNP_TOP oa_var="s_xnptop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xnp/11.5.0</XNP_TOP>
        <XNS_TOP oa_var="s_xnstop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xns/11.5.0</XNS_TOP>
        <XTR_TOP oa_var="s_xtrtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/xtr/11.5.0</XTR_TOP>
```

```
         <ZFA_TOP oa_var="s_zfatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/zfa/11.5.0</ZFA_TOP>
         <ZPB_TOP oa_var="s_zpbtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/zpb/11.5.0</ZPB_TOP>
         <ZSA_TOP oa_var="s_zsatop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/zsa/11.5.0</ZSA_TOP>
         <ZX_TOP oa_var="s_zxtop" oa_type="PROD_TOP"
oa_enabled="FALSE">/apps/oefin/appl_top/zx/11.5.0</ZX_TOP>
         <GWYUID oa_var="s_gwyuid">APPLSYSPUB/PUB</GWYUID>
         <AFSYSCSI oa_var="s_systemcsi">N/A</AFSYSCSI>
         <APPLBIN oa_var="s_applbin">bin</APPLBIN>
         <APPLDOC oa_var="s_appldoc">docs</APPLDOC>
         <APPLFRM oa_var="s_applfrm">forms</APPLFRM>
         <APPLGRAF oa_var="s_applgraf">graphs</APPLGRAF>
         <APPLIMG oa_var="s_applimg">images</APPLIMG>
         <APPLINC oa_var="s_applinc">include</APPLINC>
         <APPLLIB oa_var="s_appllib">lib</APPLLIB>
         <APPLMSG oa_var="s_applmsg">mesg</APPLMSG>
         <APPLPLS oa_var="s_applpls">plsql</APPLPLS>
         <APPLREG oa_var="s_applreg">regress</APPLREG>
         <APPLREP oa_var="s_applrep">reports</APPLREP>
         <APPLRGT oa_var="s_applrgt">regress</APPLRGT>
         <APPLRSC oa_var="s_applrsc">resource</APPLRSC>
         <APPLSAV oa_var="s_applsav">save</APPLSAV>
         <APPLSQL oa_var="s_applsql">sql</APPLSQL>
         <APPLUSR oa_var="s_applusr">usrxit</APPLUSR>
         <ADPERLPRG oa_var="s_adperlprg"
osd="unix">/apps/oefin/product/iAS/Apache/perl/bin/perl</ADPERLPRG>
         <PERL5LIB oa_var="s_perl5lib"
osd="unix">/apps/oefin/product/iAS/Apache/perl/lib/5.00503:/apps/oefin/product/iAS/Apache/
perl/lib/site_perl/5.005:/apps/oefin/appl_top/au/11.5.0/perl</PERL5LIB>
         <FORMS60_RESTRICT_ENTER_QUERY
oa_var="s_forms60_restrict_enter_query">TRUE</FORMS60_RESTRICT_ENTER_QUERY>
         <COMMON_TOP oa_var="s_com">/apps/oefin/common_top</COMMON_TOP>
      </oa_environment>
   </oa_environments>
   <oa_processes>
      <oa_process type="apache">
         <oa_process_name oa_var="s_apcname">Oracle Apache Server
OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_apcstatus">enabled</oa_process_status>
         <oa_process_log
oa_var="s_apclog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/adapcctl.txt</o
a_process_log>
         <oa_process_env>
            <oa_env_include>web_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>adovars</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_apctimeout">100</timeout>
         <ctrl_script oa_var="s_apcctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adapcctl.sh</ctrl_
script>
         <install_script oa_var="s_apcinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvapc.sh</instal
l_script>
      </oa_process>
      <oa_process type="apache_restrict">
         <oa_process_name oa_var="s_apc_restrict_name">Oracle Restricted Apache Server
OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_apc_restrict_status">disabled</oa_process_status>
         <oa_process_log
oa_var="s_apc_restrict_log">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/adapc
ctl.txt</oa_process_log>
```

```
         <oa_process_env>
            <oa_env_include>web_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>adovars</oa_env_include>
         </oa_process_env>
         <ctrl_script oa_var="s_apc_restrict_ctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adaprstctl.sh</ctr
l_script>
         <install_script oa_var="s_apc_restrict_inst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvapc.sh</instal
l_script>
      </oa_process>
      <oa_process type="apache_pls">
         <oa_process_name oa_var="s_apcname_pls">Oracle Apache Server
OEFIN_dcap-dca-oraapp01 for PL/SQL</oa_process_name>
         <oa_process_status oa_var="s_apcstatus_pls">disabled</oa_process_status>
         <oa_process_log
oa_var="s_apclog_pls">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/adapcctl.tx
t</oa_process_log>
         <oa_process_env>
            <oa_env_include>web_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_apcplstimeout">100</timeout>
         <ctrl_script oa_var="s_apcplsctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adapcctl.sh</ctrl_
script>
         <install_script oa_var="s_apcplsinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvapc.sh</instal
l_script>
      </oa_process>
      <oa_process type="tns_apps">
         <oa_process_name oa_var="s_tnsname"
osd="unix">OracleTNSListener80APPS_OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_tnsstatus">enabled</oa_process_status>
         <oa_process_log
oa_var="s_tnslog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/adalnctl.txt</o
a_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_tnstimeout">100</timeout>
         <ctrl_script oa_var="s_tnsctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adalnctl.sh</ctrl_
script>
         <install_script oa_var="s_tnsinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvalsn.sh</insta
ll_script>
      </oa_process>
      <oa_process type="tcf">
         <oa_process_name oa_var="s_tcfname">Oracle TCF SocketServer
OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_tcfstatus">disabled</oa_process_status>
         <oa_process_log
oa_var="s_tcflog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/adtcfctl.txt</o
a_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>adovars</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
```

```
         <timeout oa_var="s_tcftimeout">100</timeout>
         <ctrl_script oa_var="s_tcfctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adtcfctl.sh</ctrl_
script>
         <install_script oa_var="s_tcfinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvtcf.sh</instal
l_script>
      </oa_process>
      <oa_process type="concmgr">
         <oa_process_name
oa_var="s_concname">OracleConcMgrOEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_concstatus">disabled</oa_process_status>
         <oa_process_log
oa_var="s_conclog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/adcmctl.txt</o
a_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>adovars</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_conctimeout">1000</timeout>
         <ctrl_script_param oa_var="s_concctrl_params">diag=N wait=N</ctrl_script_param>
         <ctrl_script oa_var="s_concctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adcmctl.sh</ctrl_s
cript>
         <install_script oa_var="s_concinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvcm.sh</install
_script>
      </oa_process>
      <oa_process type="forms">
         <oa_process_name
oa_var="s_formsname">OracleFormsServer-Forms60OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_formsstatus">enabled</oa_process_status>
         <oa_process_log
oa_var="s_formslog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/f60svrm.txt</
oa_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>adovars</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_formstimeout">100</timeout>
         <ctrl_script oa_var="s_formsctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adfrmctl.sh</ctrl_
script>
         <install_script oa_var="s_formsinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvfrm.sh</instal
l_script>
      </oa_process>
      <oa_process type="reports">
         <oa_process_name
oa_var="s_reptname">OracleReportServer-Rep60_OEFIN</oa_process_name>
         <oa_process_status oa_var="s_reptstatus">disabled</oa_process_status>
         <oa_process_log
oa_var="s_repslog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/rep60_OEFIN_dc
ap-dca-oraapp01.txt</oa_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_reptstimeout">100</timeout>
```

```
         <ctrl_script oa_var="s_reptsctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adrepctl.sh</ctrl_
script>
         <install_script oa_var="s_reptsinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvrep.sh</instal
l_script>
      </oa_process>
      <oa_process type="met_cl">
         <oa_process_name oa_var="s_metcname">Oracle Metrics Client
OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_metcstatus">enabled</oa_process_status>
         <oa_process_log
oa_var="s_metclog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/adfmcctl.txt</
oa_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_metcltimeout">100</timeout>
         <ctrl_script oa_var="s_metclctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adfmcctl.sh</ctrl_
script>
         <install_script oa_var="s_metclinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvfmc.sh</instal
l_script>
      </oa_process>
      <oa_process type="met_srv">
         <oa_process_name oa_var="s_metsname">Oracle Metrics Server
OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_metsstatus">enabled</oa_process_status>
         <oa_process_log
oa_var="s_metslog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/adfmsctl.txt</
oa_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_metcsrvtimeout">100</timeout>
         <ctrl_script oa_var="s_metcsrvctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/adfmsctl.sh</ctrl_
script>
         <install_script oa_var="s_metcsrvinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adsvfms.sh</instal
l_script>
      </oa_process>
      <oa_process type="icsm_srv">
         <oa_process_name oa_var="s_icsmname">Oracle ICSM
OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_icsmstatus">disabled</oa_process_status>
         <oa_process_log
oa_var="s_icsmlog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/ieoicctl.txt</
oa_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>adovars</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_icsmtimeout">100</timeout>
         <ctrl_script oa_var="s_icsmctrl"
osd="unix">/apps/oefin/appl_top/ieo/11.5.0/admin/scripts/OEFIN_dcap-dca-oraapp01/ieoicsm.s
h</ctrl_script>
```

```
         <install_script oa_var="s_icsminst"
osd="unix">/apps/oefin/appl_top/ieo/11.5.0/admin/install/OEFIN_dcap-dca-oraapp01/ieosvicsm
.sh</install_script>
      </oa_process>
      <oa_process type="jtff_srv">
         <oa_process_name oa_var="s_jtffsname">Oracle Fulfillment Server
OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_jtffsstatus">enabled</oa_process_status>
         <oa_process_log
oa_var="s_jtffslog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/jtffmctl.txt<
/oa_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>adovars</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_jtffcsrvtimeout">100</timeout>
         <ctrl_script oa_var="s_jtffcsrvctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/jtffmctl.sh</ctrl_
script>
         <install_script oa_var="s_jtffcsrvinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/jtfsvfm.sh</instal
l_script>
         <debug_string oa_var="s_jto_debug_string">full</debug_string>
         <log_level oa_var="s_jto_log_level">9</log_level>
         <fax_enabler
oa_var="s_jto_fax_enabler">oracle.apps.jtf.fm.engine.rightfax.RfFaxEnablerImpl</fax_enable
r>
         <print_enabler
oa_var="s_jto_print_enabler">oracle.apps.jtf.fm.engine.rightfax.RfPrintEnablerImpl</print_
enabler>
         <rfjava_loc
oa_var="s_jto_rfjava_loc">/apps/oefin/common_top/java/3rdparty/RFJavaInt.zip</rfjava_loc>
         <enabler_classpath oa_var="s_jto_enabler_classpath"/>
         <show_warning oa_var="s_jto_show_warnings">false</show_warning>
         <fullfillment_serverid oa_var="s_jto_server_id">5000</fullfillment_serverid>
         <jto_classpath oa_var="s_jto_classpath"
osd="unix">.:/apps/oefin/common_top/java/jdbc111.zip:/apps/oefin/common_top/java/xmlparser
v2.zip:/apps/oefin/common_top/java:/apps/oefin/common_top/java/apps.zip:/apps/oefin/common
_top/util/java/1.4/j2sdk1.4.2_04/classes:/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_0
4/lib:/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/lib/classes.zip:/apps/oefin/commo
n_top/util/java/1.4/j2sdk1.4.2_04/lib/classes.jar:/apps/oefin/common_top/util/java/1.4/j2s
dk1.4.2_04/lib/rt.jar:/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/lib/i18n.jar:/app
s/oefin/common_top/java/3rdparty/RFJavaInt.zip:</jto_classpath>
         <start_cmd
oa_var="s_jtffstart">/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/bin/java -ms128m
-mx256m -classpath
.:/apps/oefin/common_top/java/jdbc111.zip:/apps/oefin/common_top/java/xmlparserv2.zip:/app
s/oefin/common_top/java:/apps/oefin/common_top/java/apps.zip:/apps/oefin/common_top/util/j
ava/1.4/j2sdk1.4.2_04/classes:/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/lib:/apps
/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/lib/classes.zip:/apps/oefin/common_top/util/
java/1.4/j2sdk1.4.2_04/lib/classes.jar:/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/
lib/rt.jar:/apps/oefin/common_top/util/java/1.4/j2sdk1.4.2_04/lib/i18n.jar:/apps/oefin/com
mon_top/java/3rdparty/RFJavaInt.zip:
-Dengine.LogPath=/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01
-Dengine.TempDir=/apps/oefin/common_top/temp -Dengine.CommandPort=9300
-Dengine.AOLJ.config=/apps/oefin/appl_top/fnd/11.5.0/secure/OEFIN_dcap-dca-oraapp01/oefin.
dbc -Dengine.ServerID=5000 -Ddebug=full -Dengine.LogLevel=9 -Dlog.ShowWarnings=false
-Dengine.FaxEnabler=oracle.apps.jtf.fm.engine.rightfax.RfFaxEnablerImpl
-Dengine.PrintEnabler=oracle.apps.jtf.fm.engine.rightfax.RfPrintEnablerImpl
-Dfax.TempDir=/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01
```

```
-Dprint.TempDir=/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01
oracle.apps.jtf.fm.FulfillmentServer >>
/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/jtffmctl.txt</start_cmd>
      </oa_process>
      <oa_process type="icxblkldr">
         <oa_process_name oa_var="s_icxblkname">Oracle iProcurement Bulk Loader
OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_icxblkstatus">disabled</oa_process_status>
         <oa_process_log
oa_var="s_icxblklog">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/icxblkctl.tx
t</oa_process_log>
         <oa_process_env>
            <oa_env_include>tools_home</oa_env_include>
            <oa_env_include>applsys</oa_env_include>
            <oa_env_include>adovars</oa_env_include>
            <oa_env_include>generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_icxblksrvtimeout">100</timeout>
         <ctrl_script oa_var="s_icxblksrvctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/icxblkctl.sh</ctrl
_script>
         <install_script oa_var="s_icxblksrvinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/icxblksvc.sh</inst
all_script>
      </oa_process>
      <oa_process type="disco" osd="unix">
         <oa_process_name oa_var="s_disconame" osd="unix">Oracle Discoverer services
OEFIN_dcap-dca-oraapp01</oa_process_name>
         <oa_process_status oa_var="s_discostatus" osd="unix">disabled</oa_process_status>
         <oa_process_log oa_var="s_discolog"
osd="unix">/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01/addisctl.txt</oa_proce
ss_log>
         <oa_process_env>
            <oa_env_include osd="unix">web_home</oa_env_include>
            <oa_env_include osd="unix">applsys</oa_env_include>
            <oa_env_include osd="unix">generic_service</oa_env_include>
         </oa_process_env>
         <timeout oa_var="s_discotimeout">300</timeout>
         <ctrl_script oa_var="s_discoctrl"
osd="unix">/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/addisctl.sh</ctrl_
script>
         <install_script oa_var="s_discoinst"
osd="unix">/apps/oefin/common_top/admin/install/OEFIN_dcap-dca-oraapp01/adddisctl.sh</inst
all_script>
      </oa_process>
   </oa_processes>
   <oa_customized/>
</oa_context>
```

LISTENER.ora

```
# $Header: admk80ln_ux.sql 115.27 2006/05/09 21:18:41 mmanku ship $
# LISTENER.ORA For Oracle Applications
# This file is automatically generated
APPS_OEFIN =
  (ADDRESS_LIST =
    (ADDRESS= (PROTOCOL= TCP)(Host= dcap-dca-oraapp01)(Port= 1626))
  )
SID_LIST_APPS_OEFIN =
  (SID_LIST =
    ( SID_DESC = ( SID_NAME = FNDSM )
                 ( ORACLE_HOME = /apps/oefin/product/8.0.6 )
```

```
                  ( PROGRAM = /apps/oefin/appl_top/fnd/11.5.0/bin/FNDSM )
                  (
envs='MYAPPSORA=/apps/oefin/appl_top/APPSOEFIN_dcap-dca-oraapp01.env,PATH=/usr/bin:/usr/cc
s/bin:/bin,FNDSM_SCRIPT=/apps/oefin/common_top/admin/scripts/OEFIN_dcap-dca-oraapp01/gsmst
art.sh' )
    )
    ( SID_DESC = ( SID_NAME = FNDFS  )
                  ( ORACLE_HOME = /apps/oefin/product/8.0.6 )
                  ( PROGRAM = /apps/oefin/appl_top/fnd/11.5.0/bin/FNDFS )
                  (
envs='EPC_DISABLED=TRUE,NLS_LANG=American_America.UTF8,LD_LIBRARY_PATH=/usr/dt/lib:/usr/op
enwin/lib:/apps/oefin/product/8.0.6/lib,SHLIB_PATH=/usr/lib:/usr/dt/lib:/usr/openwin/lib:/
apps/oefin/product/8.0.6/lib,LIBPATH=/usr/dt/lib:/usr/openwin/lib:/apps/oefin/product/8.0.
6/lib,APPLFSTT=OEFIN_FO;OEFIN_806_BALANCE;OEFIN;OEFIN_BALANCE,APPLFSWD=/apps/oefin/appl_to
p/admin;/apps/oefin/common_top/temp;/apps/oefin/common_top/html/oam/nonUix/launchMode/rest
ricted' )
    )
  )
# Listener general parameters
STARTUP_WAIT_TIME_APPS_OEFIN = 0
CONNECT_TIMEOUT_APPS_OEFIN = 10
TRACE_LEVEL_APPS_OEFIN = OFF
LOG_DIRECTORY_APPS_OEFIN = /apps/local/OEFIN/8.0.6/network/admin
LOG_FILE_APPS_OEFIN = APPS_OEFIN
TRACE_DIRECTORY_APPS_OEFIN = /apps/local/OEFIN/8.0.6/network/admin
TRACE_FILE_APPS_OEFIN = APPS_OEFIN
ADMIN_RESTRICTIONS_APPS_OEFIN = OFF
IFILE =
/apps/local/OEFIN/8.0.6/network/admin/OEFIN_dcap-dca-oraapp01/OEFIN_dcap-dca-oraapp01_list
ener_ifile.ora
```

TNSNAMES.ora

```
################################################################
#
# This file is automatically generated by AutoConfig.  It will be read and
# overwritten.  If you were instructed to edit this file, or if you are not
# able to use the settings created by AutoConfig, refer to Metalink document
# 165195.1 for assistance.
#
#$Header: NetServiceHandler.java 115.51 2006/10/19 09:14:49 nsugguna ship $
#
################################################################
OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=db-oefin.gslb.dcap.com)(PORT=1521))
            (CONNECT_DATA=
                (SID=OEFIN)
            )
        )
OEFIN_806_BALANCE=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=db-oefin.gslb.dcap.com)(PORT=1521))
            (CONNECT_DATA=
                (SID=OEFIN)
            )
        )
OEFIN_FO=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=db-oefin.gslb.dcap.com)(PORT=1521))
            (CONNECT_DATA=
```

```
                (SID=OEFIN)
            )
        )
OEFIN_BALANCE=
        (DESCRIPTION=
            (LOAD_BALANCE=YES)
            (FAILOVER=YES)
            (ADDRESS_LIST=
                (ADDRESS=(PROTOCOL=tcp)(HOST=db-oefin.gslb.dcap.com)(PORT=1521))
            )
            (CONNECT_DATA=
                (SID=OEFIN)
            )
        )

FNDFS_DCAP-DCA-ORAAPP02=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_DCAP-DCA-ORAAPP02.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_OEFIN_DCAP-DCA-ORAAPP02=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_OEFIN_DCAP-DCA-ORAAPP02.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )

FNDFS_DCAP-DCA-ORACM01=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORACM01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_DCAP-DCA-ORACM01.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORACM01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_OEFIN_DCAP-DCA-ORACM01=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORACM01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
```

```
        )
FNDFS_OEFIN_DCAP-DCA-ORACM01.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORACM01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )

FNDFS_DCAP-DCB-ORAAPP01=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_DCAP-DCB-ORAAPP01.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_OEFIN_DCAP-DCB-ORAAPP01=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_OEFIN_DCAP-DCB-ORAAPP01.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )

FNDFS_DCAP-DCB-ORAAPP02=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_DCAP-DCB-ORAAPP02.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_OEFIN_DCAP-DCB-ORAAPP02=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_OEFIN_DCAP-DCB-ORAAPP02.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
```

```
            )
        )

FNDFS_DCAP-DCA-ORAAPP01=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_DCAP-DCA-ORAAPP01.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_OEFIN_DCAP-DCA-ORAAPP01=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )
FNDFS_OEFIN_DCAP-DCA-ORAAPP01.dcap.com=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDFS)
            )
        )

FNDSM_DCAP-DCA-ORAAPP02_OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )
FNDSM_DCAP-DCA-ORAAPP02.dcap.com_OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )

FNDSM_DCAP-DCA-ORACM01_OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORACM01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )
FNDSM_DCAP-DCA-ORACM01.dcap.com_OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORACM01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )

FNDSM_DCAP-DCB-ORAAPP01_OEFIN=
        (DESCRIPTION=
```

```
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )
FNDSM_DCAP-DCB-ORAAPP01.dcap.com_OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )

FNDSM_DCAP-DCB-ORAAPP02_OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )
FNDSM_DCAP-DCB-ORAAPP02.dcap.com_OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCB-ORAAPP02.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )

FNDSM_DCAP-DCA-ORAAPP01_OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )
FNDSM_DCAP-DCA-ORAAPP01.dcap.com_OEFIN=
        (DESCRIPTION=
                (ADDRESS=(PROTOCOL=tcp)(HOST=DCAP-DCA-ORAAPP01.dcap.com)(PORT=1626))
            (CONNECT_DATA=
                (SID=FNDSM)
            )
        )

Rep60_OEFIN,Rep60_OEFIN.world=(ADDRESS=(PROTOCOL=tcp)(HOST=dcap-dca-oraapp01)(PORT=7000))

IFILE=/apps/local/OEFIN/8.0.6/network/admin/OEFIN_dcap-dca-oraapp01/OEFIN_dcap-dca-oraapp0
1_ifile.ora
```

Environment Files

APPSOEFIN_dcap-dca-orapp01.env

```
# $Header: APPSORA_ux.env 115.5 2003/04/01 07:38:39 isikdar ship $
# ==========================================================================
# NAME
#    APPSORA.env
#
# DESCRIPTION
#    Execute environment for Oracle and APPL_TOP
#
# NOTES
#
```

```
# HISTORY
#
# ==========================================================================
#
# ##############################################################
#
# This file is automatically generated by AutoConfig.  It will be read and
# overwritten.  If you were instructed to edit this file, or if you are not
# able to use the settings created by AutoConfig, refer to Metalink document
# 165195.1 for assistance.
#
# ##############################################################
#
# Source the custom file if it exists
customfile=/apps/oefin/appl_top/customOEFIN_dcap-dca-oraapp01.env
if [  -f $customfile ]; then
. /apps/oefin/appl_top/customOEFIN_dcap-dca-oraapp01.env
fi
. /apps/local/OEFIN/8.0.6/OEFIN_dcap-dca-oraapp01.env
. /apps/oefin/appl_top/OEFIN_dcap-dca-oraapp01.env
OEFIN_dcap-dca-oraapp01.env:
#!/bin/sh
#
# $Header: APPLSYS_ux.env 115.89 2006/08/18 06:09:16 sbandla ship $
#
# ##############################################################
#
# This file is automatically generated by AutoConfig.  It will be read and
# overwritten.  If you were instructed to edit this file, or if you are not
# able to use the settings created by AutoConfig, refer to Metalink document
# 165195.1 for assistance.
#
# ##############################################################
#
#
# The APPLFENV variable is the filename of this file.
# If you rename this file, you should change this value.
#
APPLFENV="OEFIN_dcap-dca-oraapp01.env"
export APPLFENV
#
# The CONTEXT_FILE variable stores the location of the context file.
#
CONTEXT_FILE="/apps/oefin/appl_top/admin/OEFIN_dcap-dca-oraapp01.xml"
export CONTEXT_FILE
#
# The CONTEXT_NAME variable stores the value for the current context
#
CONTEXT_NAME="OEFIN_dcap-dca-oraapp01"
export CONTEXT_NAME
#
# The PLATFORM variable is the Oracle name for this platform.
# The value below should match the value in adpltfrm.txt.
#
PLATFORM="LINUX"
export PLATFORM
#
# APPL_TOP is the top-level directory for Oracle Applications.
#
APPL_TOP="/apps/oefin/appl_top"
export APPL_TOP
#
# FNDNAM is the name of your AOL schema.
#
```

```
FNDNAM="APPS"
export FNDNAM
#
# GWYUID is the schema name and password for your public schema.
#
GWYUID="APPLSYSPUB/PUB"
export GWYUID
#
# The APPCPNAM determines how files are named by cm
# Possible values are USER and REQID.
#
APPCPNAM="REQID"
export APPCPNAM

#
# The APPLMAIL variable is needed for relinking.
# You may edit its definition, but do not remove it.
# APPLMAIL={NONE | ORACLE_INTEROFFICE}
#
APPLMAIL="NONE"
export APPLMAIL
#
# Top-level directories for all products
#
if test -f $CONTEXT_FILE; then
  allprods=`grep 'oa_type="PROD_TOP"' $CONTEXT_FILE |
     sed 's/^.*<\([A-Z_a-z0-9]*\).*oa_var[^>]*>[ ]*\([^<]*\)<.*$/\1=\2; export \1;/g'`
  # set IFS to a NEWLINE - "; needs to be at the beginning of the line
  oIFS="$IFS"
  IFS="
";
  for prod in $allprods
  do
       eval $prod
  done
  # reset the IFS to space (default)
  IFS="$oIFS"
  unset allprods #Needed for bug 3947329. This frees up space needed on AIX 4.3
else
  # This is for the corner case where somebody removed the
  # context file and needs to rebuild the context file.
  AD_TOP="/apps/oefin/appl_top/ad/11.5.0"
  export AD_TOP
  FND_TOP="/apps/oefin/appl_top/fnd/11.5.0"
  export FND_TOP
fi
#
# Get environment variable settings from fndenv.env and devenv.env
#
. /apps/oefin/appl_top/fnd/11.5.0/fndenv.env

#
# Reset PATH
#
forpath=''
for pathseg in `echo $PATH | sed "s/:/ /g"`
do
  if test -f "$pathseg/aiap"    -o -f "$pathseg/aiap60"  -o \
          -f "$pathseg/FNDLIBR" -o -f "$pathseg/f60webmx" -o \
          -f "$pathseg/adaimgr"; then
    if test -f "$pathseg/oracle"; then
      forpath="$forpath:$pathseg"
    fi
  else
```

```
    forpath="$forpath:$pathseg"
  fi
done
unzippath="/apps/oefin/common_top/util/unzip/unzip";
if test "Linux" = "HP-UX" -a \
       -x "/apps/oefin/common_top/util/unzip/unzip/unzip/unzip-RUN/opt/unzip/bin/unzip" ;
then
   unzippath="/apps/oefin/common_top/util/unzip/unzip/unzip/unzip-RUN/opt/unzip/bin";
fi
if test "Linux" = "Linux" -a \
        -x "/apps/oefin/common_top/util/unzip/unzip/unzip-5.50/unzip" ; then
   unzippath="/apps/oefin/common_top/util/unzip/unzip/unzip-5.50";
fi
PATH="/apps/oefin/appl_top/fnd/11.5.0/bin:/apps/oefin/appl_top/ad/11.5.0/bin:/apps/oefin/c
ommon_top/util/java/1.4/j2sdk1.4.2_04/bin:${unzippath}:$forpath"
export PATH
unset forpath
unset unzippath
#
# APPLDCP tells the Concurrent Manager whether you are using
# the Distributed Concurrent Processing feature.
#
APPLDCP="OFF"
export APPLDCP
#
# APPLCSF is the top-level directory in which the Concurrent Manager
# puts log and output files.
#
APPLCSF="/apps/oefin/common_top/admin"
if test "$APPLCSF" != ""
then
  export APPLCSF
fi
#
# AFSYSCSI is the CSI number
#
AFSYSCSI="N/A"
#
# APPLLOG and APPLOUT are the subdirectories in which
# the Concurrent Manager puts log and output files.
#
APPLLOG="log/OEFIN_dcap-dca-oraapp01"
export APPLLOG
APPLOUT="out/OEFIN_dcap-dca-oraapp01"
export APPLOUT
APPLRGF="/apps/oefin/common_top/rgf/OEFIN_dcap-dca-oraapp01"
export APPLRGF
#
# APPLTMP is the directory in which Oracle Applications
# temporary files are created.
#
APPLTMP="/apps/oefin/common_top/temp"
export APPLTMP
#
# APPLPTMP is the directory in which PL/SQL output files are created.
#
APPLPTMP="/usr/tmp"
export APPLPTMP
#
# Env variable that will be to set the LD_LIBRARY_PATH for Java concurrent
# programs
#
```

```
AF_LD_LIBRARY_PATH=/apps/oefin/product/iAS/lib:/apps/oefin/product/8.0.6/network/jre11/lib
/i686/native_threads:/apps/oefin/product/8.0.6/network/jre11/lib/linux/native_threads:/app
s/oefin/appl_top/cz/11.5.0/bin:${LD_LIBRARY_PATH:=}
export AF_LD_LIBRARY_PATH
#
# National Language Support environment variables
#
NLS_LANG="American_America.UTF8"
export NLS_LANG
NLS_DATE_FORMAT="DD-MON-RR"
export NLS_DATE_FORMAT
NLS_NUMERIC_CHARACTERS=".,"
export NLS_NUMERIC_CHARACTERS
NLS_SORT="BINARY"
export NLS_SORT
#
# Oracle Reports 6.0 environment variables
#
REPORTS60_TMP="/apps/oefin/common_top/temp"
export REPORTS60_TMP
#
# Reset REPORTS60_PATH
#
REPORTS60_PATH="/apps/oefin/appl_top/au/11.5.0/plsql:/apps/oefin/appl_top/fnd/11.5.0/repor
ts:/apps/oefin/appl_top/au/11.5.0/reports:/apps/oefin/appl_top/au/11.5.0/graphs:/apps/oefi
n/product/8.0.6/reports60/admin/printer"
export REPORTS60_PATH

#
# Oracle Forms 6.0 environment variables
#
PATH=$ORACLE_HOME/bin:${PATH}
export PATH
#
# Oracle Graphics 6.0 environment variables
#
#
# Reset GRAPHICS60_PATH
#
GRAPHICS60_PATH="/apps/oefin/appl_top/au/11.5.0/graphs"
export GRAPHICS60_PATH
ORAPLSQLLOADPATH="/apps/oefin/appl_top/au/11.5.0/graphs"
export ORAPLSQLLOADPATH
#
# Oracle Forms 6.0 environment variables
#
FORMS60_MAPPING="http://dcap-dca-oraapp01.dcap.com:8000/OA_TEMP"
export FORMS60_MAPPING
FORMS60_DISABLE_UNPAD_LOV="FALSE"
export FORMS60_DISABLE_UNPAD_LOV
#
#  Directory where the reports configuration file exists
#
REPORTS60_SERVER_CONFDIR="/apps/local/OEFIN/8.0.6/reports60/server"
export REPORTS60_SERVER_CONFDIR
#
# Define the Full path of the Reviver Process PID
#
FNDREVIVERPID="/apps/oefin/appl_top/fnd/11.5.0/log/reviver.sh_OEFIN_dcap-dca-oraapp01.pid"
export FNDREVIVERPID
#
# Define the variable AFCPDNR
#
TEMP_APPS_VERSION="11.5.10.2"
```

```
APPS_VERSION=`echo $TEMP_APPS_VERSION | grep 11.5 | awk -F. '{ print $1 }'`
if [ "x${APPS_VERSION}" = "x11" ]; then
  AFCPDNR="disabled"
  export AFCPDNR
fi
CNTL_BREAK="ON"
export CNTL_BREAK
FORMS60_MESSAGE_ENCRYPTION="TRUE"
export FORMS60_MESSAGE_ENCRYPTION
FORMS60_OUTPUT="/apps/oefin/common_top/temp"
export FORMS60_OUTPUT
FORMS60_SESSION="TRUE"
export FORMS60_SESSION
FORMS60_OAM_FRD="OFF"
export FORMS60_OAM_FRD
ORACLE_TERM="vt220"
export ORACLE_TERM
FORMS60_APPSLIBS="APPCORE FNDSQF APPDAYPK APPFLDR GLCORE HR_GEN HR_SPEC ARXCOVER"
export FORMS60_APPSLIBS
FORMS60_FORCE_MENU_MNEMONICS="0"
export FORMS60_FORCE_MENU_MNEMONICS
FORMS60_TRACE_PATH=/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01
export FORMS60_TRACE_PATH
#
# Reset FORMS60_PATH
#
FORMS60_PATH="/apps/oefin/appl_top/au/11.5.0/resource:/apps/oefin/appl_top/au/11.5.0/resou
rce/stub"
export FORMS60_PATH

#
# Environment variables for integration with 3rd-party products
#
#
# For integration with ILOG
#
APPL_CPLEX_LICDIR="/apps/oefin/appl_top/admin/cplex"
export APPL_CPLEX_LICDIR
#
# Add perl location to the PATH and
# set PERL5LIB for Perl
#
PERLBIN=`dirname /apps/oefin/product/iAS/Apache/perl/bin/perl`
PATH=${PERLBIN}:${PATH}
PERL5LIB=/apps/oefin/product/iAS/Apache/perl/lib/5.00503:/apps/oefin/product/iAS/Apache/pe
rl/lib/site_perl/5.005:/apps/oefin/appl_top/au/11.5.0/perl
ADPERLPRG=/apps/oefin/product/iAS/Apache/perl/bin/perl
export PATH
export PERL5LIB
export ADPERLPRG
#
# Get customer-defined environment variable settings
#
. /apps/oefin/appl_top/admin/adovars.env
#
# Applications COMMON_TOP directory
#
COMMON_TOP="/apps/oefin/common_top"
export COMMON_TOP
#
# iAS ORACLE_HOME directory
#
IAS_ORACLE_HOME="/apps/oefin/product/iAS"
export IAS_ORACLE_HOME
```

```
#
# Define the DBC file storage location
#
FND_SECURE="/apps/oefin/appl_top/fnd/11.5.0/secure/OEFIN_dcap-dca-oraapp01"
export FND_SECURE
FORMS60_WEB_CONFIG_FILE="/apps/oefin/common_top/html/bin/appsweb_OEFIN_dcap-dca-oraapp01.c
fg"
export FORMS60_WEB_CONFIG_FILE
REPORTS60_PRE="&5555"
export REPORTS60_PRE
REPORTS60_POST="&5556"
export REPORTS60_POST
FORMS60_BLOCKING_LONGLIST="FALSE"
export FORMS60_BLOCKING_LONGLIST
FORMS60_LOV_INITIAL="5000"
export FORMS60_LOV_INITIAL
FORMS60_TIMEOUT="5"
export FORMS60_TIMEOUT
FORMS60_RESTRICT_ENTER_QUERY="TRUE"
export FORMS60_RESTRICT_ENTER_QUERY
FORMS60_CATCHTERM="1"
export FORMS60_CATCHTERM
FORMS60_RTI_DIR="/apps/oefin/common_top/admin/log/OEFIN_dcap-dca-oraapp01"
export FORMS60_RTI_DIR

XML_REPORTS_XENVIRONMENT="/apps/oefin/product/8.0.6/guicommon6/tk60/admin/Tk2Motif_UTF8.rg
b"
export XML_REPORTS_XENVIRONMENT
NLS_DATE_LANGUAGE=""
export NLS_DATE_LANGUAGE
FORMS60_REJECT_GO_DISABLED_ITEM="0"
export FORMS60_REJECT_GO_DISABLED_ITEM
FORMS60_LOV_MINIMUM="1000"
export FORMS60_LOV_MINIMUM
FORMS60_LOV_WEIGHT="16"
export FORMS60_LOV_WEIGHT
FORMS60_NONBLOCKING_SLEEP="100"
export FORMS60_NONBLOCKING_SLEEP
FORMS60_OVERRIDE_ENV="NLS_LANG,NLS_NUMERIC_CHARACTERS,NLS_SORT,NLS_DATE_LANGUAGE,NLS_DATE_
FORMAT,FORMS60_USER_DATE_FORMAT,FORMS60_USER_DATETIME_FORMAT,FORMS60_OUTPUT_DATE_FORMAT,FO
RMS60_OUTPUT_DATETIME_FORMAT,FORMS60_ERROR_DATE_FORMAT,FORMS60_ERROR_DATETIME_FORMAT,FORMS
60_TZFILE,FORMS60_DATETIME_SERVER_TZ,FORMS60_DATETIME_LOCAL_TZ,FORMS60_USER_CALENDAR"
export FORMS60_OVERRIDE_ENV
FORMS60_MODULE_PATH="/apps/oefin/appl_top/fnd/11.5.0/forms"
export FORMS60_MODULE_PATH
os=`uname -s`
if [ $os = "Linux" ]; then
  if test -f /apps/oefin/appl_top/ad/11.5.0/bin/adgetlnxver.sh; then
    . /apps/oefin/appl_top/ad/11.5.0/bin/adgetlnxver.sh
  fi
fi
AFJSMARG=""
export AFJSMARG
AFJCPARG=""
export AFJCPARG
#
#Environmental variable to point to 8.0.6 configuration top in shared oracle home
#
TOOLS_CONFIG_HOME="/apps/local/OEFIN/8.0.6"
export TOOLS_CONFIG_HOME
#
#Environmental variable to point to iAS configuration top in shared oracle home
#
IAS_CONFIG_HOME="/apps/local/OEFIN/iAS"
```

```
export IAS_CONFIG_HOME
```

CSM Configuration

The following CSM configuration is available.

```
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.

```
Rule9:
      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: None
      Sticky Inactivity Timeout: (global)
      Answer Group 1: wwwin-oefin-dca
      Balance Method 1: Round Robin
      Balance Clause Options 1: DNS TTL: 5; Return Record Count: 1; Sticky Enable: No;
      Proximity Options 1: Proximity Enable: No
        RTT: (global)
        Zone: (global)
        Wait: (global)
      Answer Group 2: wwwin-oefin-dcb
      Balance Method 2: Round Robin
      Balance Clause Options 2: DNS TTL: 5; Return Record Count: 1; Sticky Enable: No;
      Proximity Options 2: Proximity Enable: No
        RTT: (global)
        Zone: (global)
        Wait: (global)
      Answer Group 3: SORRY-SERVERS
      Balance Method 3: Round Robin
      Balance Clause Options 3: DNS TTL: 5; Return Record Count: 1; Sticky Enable: No;
      Proximity Options 3: Proximity Enable: No
        RTT: (global)
        Zone: (global)
        Wait: (global)
Source Address Lists:
   List1:
      Name: Anywhere
      Owner: System
      Comments:
      Address Blocks: 0.0.0.0/0
List10:
      Name: wwwin-oefin.gslb.dcap.com
      Owner: System
      Comments: Authoritative domain on GSS
      Domains: wwwin-oefin.gslb.dcap.com
Group16:
      Name: wwwin-oefin-dca
      Type: VIP
      Owner: System
      Comments: dca answers
      Members: wwwin-oefin-dca:  Order: 0  Weight: 1  LT: 254
   Group17:
      Name: wwwin-oefin-dcb
      Type: VIP
      Owner: System
      Comments: dcb answers
      Members: wwwin-oefin-dcb:  Order: 0  Weight: 1  LT: 254
Group11:
      Name: SORRY-SERVERS
      Type: VIP
      Owner: System
      Comments: HTML Content when DCA and DCB are down and/or a DC is not ready
yet for client conns
```

```
      Members: SORRY-DCA:  Order: 0  Weight: 1  LT: 254
        SORRY-DCB:  Order: 0  Weight: 1  LT: 254
Shared KeepAlives:
   KeepAlive1:
      Primary IP Address: 101.1.6.20
      Type: KAL-AP
      Secondary IP Address:
      Destination Port:
      Host Tag:
      Path:
      Connection Termination Method:
      Number of Retries: 1
      Number of Successful Probes: 1
      Kal Options: KalAP CAPP Secure: true
   KeepAlive2:
      Primary IP Address: 200.0.0.21
      Type: KAL-AP
      Secondary IP Address:
      Destination Port:
      Host Tag:
      Path:
      Connection Termination Method:
      Number of Retries: 1
      Number of Successful Probes: 1
      Kal Options: KalAP CAPP Secure: true
```

HP Load Runner Configurations

There were 5 business processes used for this testing cycle. Procedures for each of the business cases are summarized below.

- Business Test Case 1—CRM_Manage_Role, page E-38
- Business Test Case 2—iProcurement_Add_Delete_item, page E-39
- Business Test Case 3—Create_Invoice, page E-39
- Business Test Case 4—Create_project_forms, page E-39
- Business Test Case 5—DCAP_Receivables, page E-40

Business Test Case 1—CRM_Manage_Role

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 Roles.

Step 11 Click Update.

Step 12 Logout and Close all browsers.

Business Test Case 2—iProcurement_Add_Delete_item

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_Invoice

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: payable vision operations.

Step 5 Click invoices: entry: invoices.

Step 6 Type: std, supplier --- para, Enter site, Invoice date: today's date, Invoice number: unique, Invoice amount.

Step 7 Click on distributions: Message “Transaction complete and 1 record applied and saved” is displayed at the bottom of the browser.

Step 8 Enter amount: 3 times of earlier amount.

Step 9 Click Tax code, Account.

Step 10 Click Save.

Step 11 Tax type - it will show 2 records save.

Step 12 Logout and Close all browsers.

Business Test Case 4—Create_project_forms

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

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 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.

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.

Volume details from the filer point of view follow:

```
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

All four Oracle database servers are HP Compaq DL380 G4s with 4 GB of RAM and 2 dual-core Intel Xeon 3.20 GHz 64-bit (x86_64) CPUs. Each server boots from an internal RAID drive. Each server has 2 internal Broadcom GigE NICs, one for private network connectivity and one for test network connectivity.PFPF The private network is used for cluster control traffic.

Servers dcap-dca-oradb01 and dcap-dca-oradb02 are in DCa and comprise the primary active/passive Oracle database cluster. Servers dcap-dcb-oradb01 and dcap-dcb-oradb02 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. The cluster software used is RedHat Cluster Suite.

Each database has 2 FC SAN ports provided by QLogic QLA2460 HBAs. The firmware version is 4.00.18 and the driver version is 8.01.04-d7.

Each server has 2 redundant FC paths to 3 data LUNs containing the Exchange database. There are 5 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 Oracle data was distributed over the 5 LUNs as follows:

- /oracle/admin/OEFIN: configuration and log files
- /oracle/archive/OEFIN/fs01: archive log files
- /oracle/oradata/OEFIN/fs01 and /oracle/oradata/OEFIN/fs02: data files
- /oracle/oradata/OEFIN/redo01: online redo logs

The Oracle executable code was provided by a NetApp NAS volume local to each data center:

- data center A: dcap-dca-nas.dcap.com:/vol/dca_oraapp_product mounted on /oracle/product
- data center B: dcap-dcb-nas.dcap.com:/vol/dcb_oraapp_product mounted on /oracle/product

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 were not used in testing.

The multipath software was native Linux MPIO, also called dm-multipath or device-mapper-multipath. For EMC and HP, the stock Linux functionality was used. For NetApp, the host attach kit was used (which provides special path selector software required when the filer cluster is running single system image mode).

The "/sbin/multipath –l" output:

```
[root@dcap-dca-oradb01-m ~]# multipath -l
SEMC_____SYMMETRIX______30032003F000
[size=67 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:4   sdap 66:144 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:4   sdk  8:160  [active][ready]
360a98000486e5366535a422f4a466150
[size=60 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:0:0   sdaf 65:240 [active][ready]
 \_ 0:0:0:0   sda  8:0    [active][ready]
SEMC_____SYMMETRIX______3003200EB000
[size=16 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:11  sdaw 67:0   [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:11  sdr  65:16  [active][ready]
SEMC_____SYMMETRIX______300320118000
[size=120 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:1   sdam 66:96  [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:1   sdh  8:112  [active][ready]
SEMC_____SYMMETRIX______3003200E9000
[size=16 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:6   sdar 66:176 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:6   sdm  8:192  [active][ready]
360060e8014439400000014394000000a2
[size=100 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 0:0:2:6   sdac 65:192 [active][ready]
\_ round-robin 0 [enabled]
 \_ 1:0:2:6   sdbh 67:176 [active][ready]
360060e8014439400000014394000000000
[size=50 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:2:0   sdbb 67:80  [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:2:0   sdw  65:96  [active][ready]
360a98000486e5366535a426d78467563
[size=100 MB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
```

```
 \_ 1:0:0:5   sdak 66:64  [active][ready]
 \_ 0:0:0:5   sdf  8:80   [active][ready]
360a98000486e5366535a422f4a633774
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:0:4   sdaj 66:48  [active][ready]
 \_ 0:0:0:4   sde  8:64   [active][ready]
360a98000486e5366535a422f4a626373
[size=10 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:0:3   sdai 66:32  [active][ready]
 \_ 0:0:0:3   sdd  8:48   [active][ready]
360060e8014439400000014394000000a1
[size=100 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 0:0:2:5   sdab 65:176 [active][ready]
\_ round-robin 0 [enabled]
 \_ 1:0:2:5   sdbg 67:160 [active][ready]
SEMC_____SYMMETRIX______3003200EA000
[size=16 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:7   sdas 66:192 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:7   sdn  8:208  [active][ready]
360a98000486e5366535a426d78724737
[size=100 MB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:0:6   sdal 66:80  [active][ready]
 \_ 0:0:0:6   sdg  8:96   [active][ready]
360060e8014439400000014394000000046
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 0:0:2:4   sdaa 65:160 [active][ready]
\_ round-robin 0 [enabled]
 \_ 1:0:2:4   sdbf 67:144 [active][ready]
360a98000486e5366535a422f4a617936
[size=60 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:0:2   sdah 66:16  [active][ready]
 \_ 0:0:0:2   sdc  8:32   [active][ready]
SEMC_____SYMMETRIX______30032002C000
[size=2 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:241 sdba 67:64  [active][ready]
360060e8014439400000014394000000008
[size=50 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:2:2   sdbd 67:112 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:2:2   sdy  65:128 [active][ready]
360060e8014439400000014394000000042
[size=10 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:2:3   sdbe 67:128 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:2:3   sdz  65:144 [active][ready]
SEMC_____SYMMETRIX______300320043000
[size=67 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:5   sdaq 66:160 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:5   sdl  8:176  [active][ready]
SEMC_____SYMMETRIX______300320047000
[size=67 GB][features="0"][hwhandler="0"]
```

```
\_ round-robin 0 [active]
 \_ 1:0:1:8   sdat 66:208 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:8   sdo  8:224  [active][ready]
SEMC_____SYMMETRIX______3003200F3000
[size=16 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:18  sdaz 67:48  [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:18  sdu  65:64  [active][ready]
SEMC_____SYMMETRIX______30032002B000
[size=2 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 0:0:1:241 sdv  65:80  [active][ready]
SEMC_____SYMMETRIX______3003200EC000
[size=16 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:12  sdax 67:16  [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:12  sds  65:32  [active][ready]
SEMC_____SYMMETRIX______300320053000
[size=67 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:9   sdau 66:224 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:9   sdp  8:240  [active][ready]
SEMC_____SYMMETRIX______300320119000
[size=120 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:2   sdan 66:112 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:2   sdi  8:128  [active][ready]
SEMC_____SYMMETRIX______300320057000
[size=67 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:10  sdav 66:240 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:10  sdq  65:0   [active][ready]
360060e8014439400000143940000000 4
[size=50 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:2:1   sdbc 67:96  [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:2:1   sdx  65:112 [active][ready]
360060e8014439400000143940000006 8
[size=46 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 0:0:2:8   sdae 65:224 [active][ready]
\_ round-robin 0 [enabled]
 \_ 1:0:2:8   sdbj 67:208 [active][ready]
SEMC_____SYMMETRIX______30032003B000
[size=67 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:3   sdao 66:128 [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:3   sdj  8:144  [active][ready]
SEMC_____SYMMETRIX______3003200F2000
[size=16 GB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 1:0:1:17  sday 67:32  [active][ready]
\_ round-robin 0 [enabled]
 \_ 0:0:1:17  sdt  65:48  [active][ready]
360a98000486e5366535a422f4a61545a
[size=60 GB][features="1 queue_if_no_path"][hwhandler="0"]
```

```
\_ round-robin 0 [active]
 \_ 1:0:0:1   sdag 66:0   [active][ready]
 \_ 0:0:0:1   sdb  8:16   [active][ready]
360060e8014439400000014394000000067
[size=46 MB][features="0"][hwhandler="0"]
\_ round-robin 0 [active]
 \_ 0:0:2:7   sdad 65:208 [active][ready]
\_ round-robin 0 [enabled]
 \_ 1:0:2:7   sdbi 67:192 [active][ready]
```

The /etc/multipath.conf file

```
defaults {
       multipath_tool           "/sbin/multipath -v0"
       udev_dir                 /dev
       polling_interval         10
       default_selector         "round-robin 0"
       default_path_grouping_policy    multibus
       default_getuid_callout   "/sbin/scsi_id -g -u -s /block/%n"
       default_prio_callout     "/bin/true"
       default_features         "0"
       rr_wmin_io               100
       failback                 immediate
}
devices {
    device {
        vendor                  "EMC     "
        product                 "SYMMETRIX     "
    }
    device {
        vendor                  "HP      "
        product                 "OPEN-V        "
    }
    device {
        vendor                  "HP      "
        product                 "OPEN-V-CM     "
    }
    device {
        vendor                  "EMC     "
        product                 "SYMMETRIX     "
    }
    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
    }
}
```

Servers were configured with Logical Volume Manager 2 (LVM2) in clustered mode providing storage from all three vendors to each host at the same time (although the OEFIN database was only up on one vendor at a time).

The volume structure incorporated a basic standard designed to provide scalability and portability of an individual database between hosts. Three volume groups were used:

1. vgXoracle (where "X" = "E" for EMC, "N" for NetApp, and "H" for HP): mount point for other Oracle logical volumes. (Experience has shown that having a small "sacrificial" /oracle mount point with separate file systems for data keeps a rogue program from filling up data file systems and causing problems.)
2. vgXYOEFIN (where "X" = "E" for EMC, "N" for NetApp, and "H" for HP and "Y" = "S" for synchronous replication): configuration and log files.
3. vgdbXYOEFIN (where "X" = "E" for EMC, "N" for NetApp, and "H" for HP and "Y" = "S" for synchronous replication): archive, online redo, and data files.

The volume naming structure provides easy additions of data file and archive file systems. The volumes are relatively small, because the Oracle Image database (which is the origin of OEFIN) is not large and a backup/recovery scheme involving tape drives benefits from having multiple relatively small file systems versus few relatively big file systems (to allow maximum parallelism of data streams).

Excerpt from /etc/fstab showing NetApp storage (including NAS) for dcap-dca-oradb01

```
/dev/vgNoracle/lvoracle /oracle ext3 defaults 1 3
/dev/vgdbNSOEFIN/lvredo01 /oracle/oradata/OEFIN/redo01 ext3 defaults 1 3
/dev/vgdbNSOEFIN/lvdb02  /oracle/oradata/OEFIN/fs02 ext3 defaults 1 3
/dev/vgdbNSOEFIN/lvdb01  /oracle/oradata/OEFIN/fs01 ext3 defaults 1 3
/dev/vgdbNSOEFIN/lvarchive01 /oracle/archive/OEFIN/fs01 ext3 defaults 1 3
/dev/vgNSOEFIN/lvoraadmin /oracle/admin/OEFIN ext3 defaults 1 3
dcap-dca-nas.dcap.com:/vol/dca_oraapp_product /oracle/product nfs
nfsvers=2,tcp,hard,intr,rsize=32768,wsize=32768 0 0
```

Key LVM Structures

```
[root@dcap-dca-oradb01-m ~]# vgs -a
  VG          #PV #LV #SN Attr   VSize   VFree
  vgESCRM01     1   1   0 wz--nc  67.43G    0
  vgESOEFIN     1   1   0 wz--nc  16.86G    0
  vgEoracle     1   1   0 wz--n- 116.00M    0
  vgHSOEFIN     1   1   0 wz--nc  10.19G    0
  vgHoracle     1   1   0 wz--n-  96.00M    0
  vgNSOEFIN     1   1   0 wz--nc  10.00G    0
  vgNoracle     1   1   0 wz--n-  96.00M    0
  vgdbESOEFIN   4   4   0 wz--nc 219.16G    0
  vgdbHSOEFIN   4   4   0 wz--nc 163.02G    0
  vgdbNSOEFIN   4   4   0 wz--nc 189.98G    0
[root@dcap-dca-oradb01-m ~]# lvs
  LV          VG          Attr   LSize   Origin Snap%  Move Log Copy%
  lvoraadmin  vgESOEFIN   -wi-a-  16.86G
  lvoracle    vgEoracle   -wi-a- 116.00M
  lvoraadmin  vgHSOEFIN   -wi-a-  10.19G
  lvoracle    vgHoracle   -wi-a-  96.00M
  lvoraadmin  vgNSOEFIN   -wi-ao  10.00G
  lvoracle    vgNoracle   -wi-ao  96.00M
  lvarchive01 vgdbESOEFIN -wi-a-  67.43G
  lvdb01      vgdbESOEFIN -wi-a-  67.43G
  lvdb02      vgdbESOEFIN -wi-a-  67.43G
  lvredo01    vgdbESOEFIN -wi-a-  16.86G
  lvarchive01 vgdbHSOEFIN -wi-a-  50.95G
  lvdb01      vgdbHSOEFIN -wi-a-  50.95G
  lvdb02      vgdbHSOEFIN -wi-a-  50.95G
  lvredo01    vgdbHSOEFIN -wi-a-  10.19G
  lvarchive01 vgdbNSOEFIN -wi-ao  60.00G
  lvdb01      vgdbNSOEFIN -wi-ao  60.00G
  lvdb02      vgdbNSOEFIN -wi-ao  60.00G
```

```
  lvredo01    vgdbNSOEFIN -wi-ao  10.00G
[root@dcap-dca-oradb02-m ~]# lvs -o +devices
 LV      VG      Attr  LSize  Origin Snap% Move Log Copy% Devices

  lvoraadmin  vgESOEFIN   -wi-a-  16.86G
/dev/mapper/SEMC_____SYMMETRIX______3003200EA000(0)
  lvoracle    vgEoracle   -wi-a- 116.00M
/dev/mapper/SEMC_____SYMMETRIX______300320118000(0)
  lvoraadmin  vgHSOEFIN   -wi-a-  10.19G
/dev/mapper/360060e8014439400000014394000000046(0)
  lvoraadmin  vgNSOEFIN   -wi-a-  10.00G
/dev/mapper/360a98000486e5366535a422f4a633774(0)
  lvoracle    vgNoracle   -wi-a-  96.00M
/dev/mapper/360a98000486e5366535a426d78467563(0)
  lvarchive01 vgdbESOEFIN -wi-a-  67.43G
/dev/mapper/SEMC_____SYMMETRIX______300320043000(0)
  lvdb01      vgdbESOEFIN -wi-a-  67.43G
/dev/mapper/SEMC_____SYMMETRIX______30032003B000(0)
  lvdb02      vgdbESOEFIN -wi-a-  67.43G
/dev/mapper/SEMC_____SYMMETRIX______30032003F000(0)
  lvredo01    vgdbESOEFIN -wi-a-  16.86G
/dev/mapper/SEMC_____SYMMETRIX______3003200E9000(0)
  lvarchive01 vgdbHSOEFIN -wi-a-  50.95G
/dev/mapper/360060e8014439400000014394000000008(0)
  lvdb01      vgdbHSOEFIN -wi-a-  50.95G
/dev/mapper/360060e8014439400000014394000000000(0)
  lvdb02      vgdbHSOEFIN -wi-a-  50.95G
/dev/mapper/360060e8014439400000014394000000004(0)
  lvredo01    vgdbHSOEFIN -wi-a-  10.19G
/dev/mapper/360060e8014439400000014394000000042(0)
  lvarchive01 vgdbNSOEFIN -wi-a-  60.00G
/dev/mapper/360a98000486e5366535a422f4a617936(0)
  lvdb01      vgdbNSOEFIN -wi-a-  60.00G
/dev/mapper/360a98000486e5366535a422f4a466150(0)
  lvdb02      vgdbNSOEFIN -wi-a-  60.00G
/dev/mapper/360a98000486e5366535a422f4a61545a(0)
  lvredo01    vgdbNSOEFIN -wi-a-  10.00G
/dev/mapper/360a98000486e5366535a422f4a626373(0)
```

The /etc/lvm/lvm.conf file

```
devices {
    dir = "/dev"
    scan = [ "/dev" ]
    filter = [ "r|^/dev/mapper/.*-lv.*|", "a|^/dev/mapper/.*|", "r/.*/" ]
    cache = "/etc/lvm/.cache"
    write_cache_state = 0
    types = [ "device-mapper", 1 ]
    sysfs_scan = 1
    md_component_detection = 1
}
log {
    verbose = 0
    syslog = 1
    file = "/var/log/lvm2.log"
    overwrite = 0
    level = 6

    indent = 1
    command_names = 0
    prefix = "  "
}
backup {
    backup = 1
```

```
    backup_dir = "/etc/lvm/backup"
    archive = 1
    archive_dir = "/etc/lvm/archive"

    retain_min = 10
    retain_days = 30
}
shell {
    history_size = 100
}
global {
    library_dir = "/usr/lib64"
    locking_library = "liblvm2clusterlock.so"

    umask = 077
    test = 0
    activation = 1
    proc = "/proc"
    locking_type = 2
    locking_dir = "/var/lock/lvm"
}
activation {
    missing_stripe_filler = "/dev/ioerror"
    reserved_stack = 256
    reserved_memory = 8192
    process_priority = -18
    mirror_region_size = 512
    mirror_log_fault_policy = "allocate"
    mirror_device_fault_policy = "remove"
}
```

SAN Storage Details

The following SAN LUN details for dcap-dca-oradb01 (other hosts are similar) are available:

- EMC, page E-48
- NetApp, page E-49
- HP, page E-49

EMC

```
[root@dcap-san-hst-06 ~]# /usr/symcli/bin/symrdf -sid 320 -rdfg 11 -nop -f
/devinfo/storage/emc/dcap-dca-oradb_11_sync.rdf query
Symmetrix ID                       : 000190300320
Remote Symmetrix ID                : 000190300321
RDF (RA) Group Number              : 11 (0A)

       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     003B RW       0        0 RW 003B WD       0        0 S..   Synchronized
N/A     003F RW       0        0 RW 003F WD       0        0 S..   Synchronized
N/A     0043 RW       0        0 RW 0043 WD       0        0 S..   Synchronized
N/A     00E9 RW       0        0 RW 00E9 WD       0        0 S..   Synchronized
N/A     00EA RW       0        0 RW 00EA 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

```
dcap-netapp-A1> vol status OEFIN_1
         Volume State      Status            Options
        OEFIN_1 online     raid_dp, flex     nosnap=on, create_ucode=on,
                                             fs_size_fixed=on
                Containing aggregate: 'aggr0'
dcap-netapp-A1> vol status OEFIN_2
         Volume State      Status            Options
        OEFIN_2 online     raid_dp, flex     nosnap=on, create_ucode=on,
                                             fs_size_fixed=on
                Containing aggregate: 'aggr1'
dcap-netapp-A1> vol status OEFIN_3
         Volume State      Status            Options
        OEFIN_3 online     raid_dp, flex     nosnap=on, create_ucode=on,
                                             fs_size_fixed=on
                Containing aggregate: 'aggr2'
dcap-netapp-A1> lun show
        /vol/OEFIN_1/lun0             60g (64424509440)   (r/w, online, mapped)
        /vol/OEFIN_1/lun3             10g (10737418240)   (r/w, online, mapped)
        /vol/OEFIN_2/lun1             60g (64424509440)   (r/w, online, mapped)
        /vol/OEFIN_2/lun4             10g (10737418240)   (r/w, online, mapped)
        /vol/OEFIN_3/lun2             60g (64424509440)   (r/w, online, mapped)
dcap-netapp-A1> lun show -m
LUN path                            Mapped to          LUN ID  Protocol
----------------------------------------------------------------------
/vol/OEFIN_1/lun0                   dcap-dca-oradb01        0       FCP
                                    dcap-dca-oradb02        0       FCP
/vol/OEFIN_1/lun3                   dcap-dca-oradb01        3       FCP
                                    dcap-dca-oradb02        3       FCP
/vol/OEFIN_2/lun1                   dcap-dca-oradb01        1       FCP
                                    dcap-dca-oradb02        1       FCP
/vol/OEFIN_2/lun4                   dcap-dca-oradb01        4       FCP
                                    dcap-dca-oradb02        4       FCP
/vol/OEFIN_3/lun2                   dcap-dca-oradb01        2       FCP
                                    dcap-dca-oradb02        2       FCP
dcap-netapp-A1> snapmirror status
Snapmirror is on.Source                             Destination
State          Lag        Status
dcap-netapp-A1:OEFIN_1                       dcap-netapp-B1:OEFIN_1         Source
-          In-sync
dcap-netapp-A1:OEFIN_2                       dcap-netapp-B1:OEFIN_2         Source
-          In-sync
dcap-netapp-A1:OEFIN_3                       dcap-netapp-B1:OEFIN_3         Source
-          In-sync
```

HP

```
[root@dcap-san-hst-10 ~]# /usr/bin/pairdisplay -I0 -g dcap-dca-oradb-s -l -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-dca-oradb-s        sync-dca-oradb-0(L) (CL5-B , 0,   0)82836     0.P-VOL PAIR NEVER
,82931     0 -
```

```
dcap-dca-oradb-s        sync-dca-oradb-1(L) (CL5-B , 0,   1)82836     4.P-VOL PAIR NEVER
,82931     4 -
dcap-dca-oradb-s        sync-dca-oradb-2(L) (CL5-B , 0,   2)82836     8.P-VOL PAIR NEVER
,82931     8 -
dcap-dca-oradb-s        sync-dca-oradb-3(L) (CL5-B , 0,   3)82836    42.P-VOL PAIR NEVER
,82931    42 -
dcap-dca-oradb-s        sync-dca-oradb-4(L) (CL5-B , 0,   4)82836    46.P-VOL PAIR NEVER
,82931    46 –
[root@dcap-san-hst-12 ~]# /usr/bin/pairdisplay -I0 -g dcap-dcb-oradb-s -l -fx
Group   PairVol(L/R) (Port#,TID, LU),Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
dcap-dcb-oradb-s        sync-dcb-oradb-0(L) (CL5-B , 0,   0)82931     0.S-VOL PAIR NEVER
,-----     0 -
dcap-dcb-oradb-s        sync-dcb-oradb-1(L) (CL5-B , 0,   1)82931     4.S-VOL PAIR NEVER
,-----     4 -
dcap-dcb-oradb-s        sync-dcb-oradb-2(L) (CL5-B , 0,   2)82931     8.S-VOL PAIR NEVER
,-----     8 -
dcap-dcb-oradb-s        sync-dcb-oradb-3(L) (CL5-B , 0,   3)82931    42.S-VOL PAIR NEVER
,-----    42 -
dcap-dcb-oradb-s        sync-dcb-oradb-4(L) (CL5-B , 0,   4)82931    46.S-VOL PAIR NEVER
,-----    46 -
```

APPENDIX F

Exchange Configuration Details

This appendix provides 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 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.

Each has 2 FC SAN ports with either 2 QLogic QLE2460 or Emulex LP10000ExDC-E HBAs.

The breakdown is as follows:

```
- LP10000ExDC-E: dcap-xchng-a1, dcap-xchng-b1
 - Firmware: 1.91A5
 - Driver: 5-1.20A3
- QLE2460: 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 4.6.1 (build 5)
- 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 for automatic DNS updates by Windows Exchange and Domain Controller servers. 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).

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.

Below are the configuration files for the master DNS servers. These show the zone files available to the secondary DNS servers.

dcap-dca-ns1

/etc/named.conf

```
// Default named.conf generated by install of bind-9.2.4-16.EL4
options {
        directory "/var/named";
        dump-file "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
        listen-on {
                10.0.5.168;
                101.1.33.12;
                127.0.0.1;
        };
        forwarders {
        };
```

```
};
acl "xchng-servers" {
  101.1.33.19; 101.1.33.20; 201.1.33.19; 201.1.33.20;
};
acl "dc-servers" {
  101.1.33.14; 201.1.33.14;
};
acl "slave-dns-servers" {
  101.1.33.14; 201.1.33.14; 10.0.10.2; 10.0.20.2; 10.0.30.2;
};
controls {
  inet 127.0.0.1 allow { localhost; 10.0.5.168; } keys { rndc_key; };
};
key "rndc_key" {
        algorithm hmac-md5;
        secret "c3Ryb25nIGVub3VnaCBmb3IgYSBtYW4gYnV0IG1hZGUgZm9yIGEgd29tYW4K";
};
zone "0.0.127.in-addr.arpa"{
        type master;
        file "named.local";
};
zone "dcap.com" {
        type master;
        file "dcap.com";
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "_msdcs.dcap.com" {
        type master;
        file "_msdcs.dcap.com";
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "_sites.dcap.com" {
        type master;
        file "_sites.dcap.com";
        allow-update { dc-servers; xchng-servers; 127.0.0.1; 10.0.5.168; 101.1.33.12; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "_tcp.dcap.com" {
        type master;
        file "_tcp.dcap.com";
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "_udp.dcap.com" {
        type master;
        file "_udp.dcap.com";
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "ForestDnsZones.dcap.com" {
        type master;
        file "ForestDnsZones.dcap.com";
```

```
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "5.0.10.in-addr.arpa" {
        type master;
        file "10.0.5.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 127.0.0.1; 10.0.5.168; 101.1.33.12; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "33.1.101.in-addr.arpa" {
        type master;
        file "101.1.33.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "33.1.201.in-addr.arpa" {
        type master;
        file "201.1.33.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "34.1.101.in-addr.arpa" {
        type master;
        file "101.1.34.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "34.1.201.in-addr.arpa" {
        type master;
        file "201.1.34.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "10.0.10.in-addr.arpa" {
        type master;
        file "10.0.10.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "20.0.10.in-addr.arpa" {
        type master;
        file "10.0.20.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "30.0.10.in-addr.arpa" {
        type master;
        file "10.0.30.x";
```

```
        notify no;
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
include "/etc/rndc.key";

=== END OF FILE ===
```

dcap-dcb-ns1

/etc/named.conf

```
// Default named.conf generated by install of bind-9.2.4-16.EL4
options {
        directory "/var/named";
        dump-file "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
        listen-on {
                10.0.5.169;
                201.1.33.12;
                127.0.0.1;
        };
        forwarders {
        };

};

acl "xchng-servers" {
  101.1.33.19; 101.1.33.20; 201.1.33.19; 201.1.33.20;
};

acl "dc-servers" {
  101.1.33.14; 201.1.33.14;
};

acl "slave-dns-servers" {
  101.1.33.14; 201.1.33.14; 10.0.10.2; 10.0.20.2; 10.0.30.2;
};

controls {
  inet 127.0.0.1 allow { localhost; 10.0.5.169; } keys { rndc_key; };
};

key "rndc_key" {
        algorithm hmac-md5;
        secret "c3Ryb25nIGVub3VnaCBmb3IgYSBtYW4gYnV0IG1hZGUgZm9yIGEgd29tYW4K";
};


zone "0.0.127.in-addr.arpa"{
        type master;
        file "named.local";
};
zone "dcap.com" {
        type master;
        file "dcap.com";
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
```

```
zone "_msdcs.dcap.com" {
        type master;
        file "_msdcs.dcap.com";
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "_sites.dcap.com" {
        type master;
        file "_sites.dcap.com";
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "_tcp.dcap.com" {
        type master;
        file "_tcp.dcap.com";
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "_udp.dcap.com" {
        type master;
        file "_udp.dcap.com";
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "ForestDnsZones.dcap.com" {
        type master;
        file "ForestDnsZones.dcap.com";
        allow-update { dc-servers; xchng-servers; 10.0.5.168; 101.1.33.12; };
        notify no;
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "5.0.10.in-addr.arpa" {
        type master;
        file "10.0.5.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 127.0.0.1; 201.1.33.12; 10.0.5.169; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "33.1.101.in-addr.arpa" {
        type master;
        file "101.1.33.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "33.1.201.in-addr.arpa" {
        type master;
        file "201.1.33.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
```

```
zone "34.1.101.in-addr.arpa" {
        type master;
        file "101.1.34.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "34.1.201.in-addr.arpa" {
        type master;
        file "201.1.34.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "10.0.10.in-addr.arpa" {
        type master;
        file "10.0.10.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "20.0.10.in-addr.arpa" {
        type master;
        file "10.0.20.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
zone "30.0.10.in-addr.arpa" {
        type master;
        file "10.0.30.x";
        notify no;
        allow-update { dc-servers; xchng-servers; 201.1.33.12; 10.0.5.169; };
        allow-query { any; };
        allow-transfer { slave-dns-servers; };
};
include "/etc/rndc.key";

=== END OF FILE ===
```

Storage Details

Here are path and LUN details for dcap-xchng-a1 (other hosts are similar):

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

For information about the hardware device configurations used in data center testing, please see the platform-specific configuration information in other sections of this document.

Disaster recovery steps required to perform data center failover and failback are in the following sections.

- Failover Overview, page G-1
- Failback Overview, page G-3

Failover Overview

Step 1 Start application transactions and verify environment.

- 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 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 commands:

```
/usr/symcli/bin/symrdf -sid 321 -force -rdfg 11 -nop -f
/devinfo/storage/emc/dcap-dca-oradb_11_sync.rdf failover

 /usr/symcli/bin/symrdf -sid 321 -force -rdfg 13 -nop -f
/devinfo/storage/emc/dcap-dca-xchg_13_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")
```

```
 /usr/symcli/bin/symrdf -sid 321 -rdfg 11 -nop -f
/devinfo/storage/emc/dcap-dca-oradb_11_sync.rdf query (check for a status of "Failed
Over")
```

Here are the configuration file contents:

```
/devinfo/storage/emc/dcap-dca-oradb_11_sync.rdf:
 3b 3b
 3f 3f
 43 43
 e9 e9
 ea ea

 /devinfo/storage/emc/dcap-dca-xchg_13_sync.rdf
 63 63
 67 67
 f5 f5
```

HP

Using the Command View GUI or HORCM CLI on the failover frame, delete the synchronous CA pairs with a "pairsplit-S" command (without the force option enabled).

NetApp

On the failover filer, dcap-netapp-b1, do:

```
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
 snapmirror quiesce dcap-netapp-B1:OEFIN_1
 snapmirror break dcap-netapp-B1:OEFIN_1
 snapmirror quiesce dcap-netapp-B1:OEFIN_2
 snapmirror break dcap-netapp-B1:OEFIN_2
 snapmirror quiesce dcap-netapp-B1:OEFIN_3
 snapmirror break dcap-netapp-B1:OEFIN_3
```

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 domain account for the Exchange Virtual Server.
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 datacenter. If not, fix it.
6. Verify DNS points to the correct address on all three branch DNS secondary servers; manually reload zone files (in the DNS application, open the Forward Lookup Zones, click "dcap.com", right-click "dcap.com", then choose "Transer From Master" and then "Reload") and purge DNS cache ("ipconfig /flushdns") as needed.
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.

Set the environment for the database.

```
sqlplus " / as sysdba"; startup
./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 application 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 datacenter failover RTO and RPO.

Failback Overview

Note Failback assumes failover has already been done.

Step 1 Start application transactions and verify environment.

- 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 2 Ensure the primary datacenter 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.) NOTE: the failback method used in this test does require application downtime for EMC and NetApp, so this step should be skipped when using storage from those vendors. For HP, since resynchronization requires copying all the data volumes (not just changes) back to DCa, this step can and should be done before the planned failback outage window to avoid excessive downtime. Starting the resynchronization does not require application downtime in the failover data center (although completing it does).

Step 3 Ensure the primary datacenter applications, including the CSM 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 commands:

```
 /usr/symcli/bin/symrdf -sid 321 -force -rdfg 11 -nop -f
/devinfo/storage/emc/dcap-dca-oradb_11_sync.rdf failback

 /usr/symcli/bin/symrdf -sid 321 -force -rdfg 13 -nop -f
/devinfo/storage/emc/dcap-dca-xchg_13_sync.rdf failback

 /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
"Synchronized")

 /usr/symcli/bin/symrdf -sid 321 -rdfg 11 -nop -f
/devinfo/storage/emc/dcap-dca-oradb_11_sync.rdf query" (check for a status of
"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_11_sync.rdf:
 3b 3b
 3f 3f
 43 43
 e9 e9
 ea ea

 /devinfo/storage/emc/dcap-dca-xchg_13_sync.rdf
 63 63
 67 67
 f5 f5
```

HP

1. Verify the pairs established in step 2 are in PAIR status using HORCM or the Command View GUI and the applications are completely offline.
2. Using the Command View GUI or HORCM CLI on the failover frame, delete the pair for each device using a fiber channel path with a "pairsplit-S" command without force enabled.

This method requires a complete copy of the data in DCb to be copied back to DCa. That's the reason for creating the pairs in step 3 before the outage window.

NetApp

On filer dcap-netapp-a1 (original source filer), do:

```
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
 snapmirror resync -f -S dcap-netapp-b1-e0c:OEFIN_1 -w dcap-netapp-A1:OEFIN_1
 snapmirror break dcap-netapp-A1:OEFIN_1
 snapmirror resync -f -S dcap-netapp-b1-e0c:OEFIN_2 -w dcap-netapp-A1:OEFIN_2
 snapmirror break dcap-netapp-A1:OEFIN_2
 snapmirror resync -f -S dcap-netapp-b1-e0c:OEFIN_3 -w dcap-netapp-A1:OEFIN_3
 snapmirror break dcap-netapp-A1:OEFIN_3
```

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 on both master DNS servers and fix as needed to point to the primary data center IP address.
5. Verify DNS points to the correct address on all three branch DNS secondary servers; manually reload zone files (DNS, click "Forward Zones" then "dcap.com", right-click on "dcap.com", then do Transfer From Master" and "Reload") and purge DNS cache as needed ("ipconfig /flushdns").
6. Ensure the branch Outlook users begin to receive email again. Check for email until the queues are clear on both sending hosts.
7. 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.

Set the environment for the database.

```
sqlplus " / as sysdba"; startup
./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 Datacenters A & B.

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 the Command View GUI or HORCM CLI on the primary frame, do a paircreate for each device. Be sure to use the correct fiber channel path.
- 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:OEFIN_1
 snapmirror release OEFIN_1 dcap-netapp-A1:OEFIN_1
 snapmirror resync -f -w dcap-netapp-B1:OEFIN_2
 snapmirror release OEFIN_2 dcap-netapp-A1:OEFIN_2
 snapmirror resync -f -w dcap-netapp-B1:OEFIN_3
 snapmirror release OEFIN_3 dcap-netapp-A1:OEFIN_3
```

Step 12 Determine the latest RTO and RPO of all applications. This is the datacenter 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:

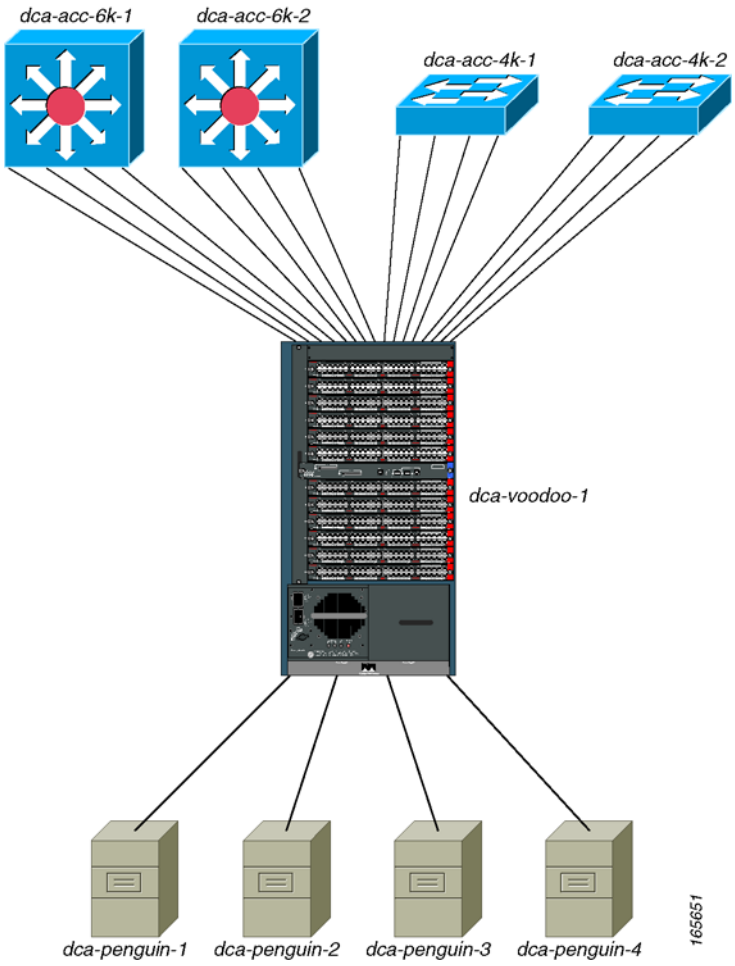
- Dual AMD Opteron processors @ 2600 MHz, w/1 GB onboard cache
- 4 GB DDR RAM
- Red Hat Enterprise Linux

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

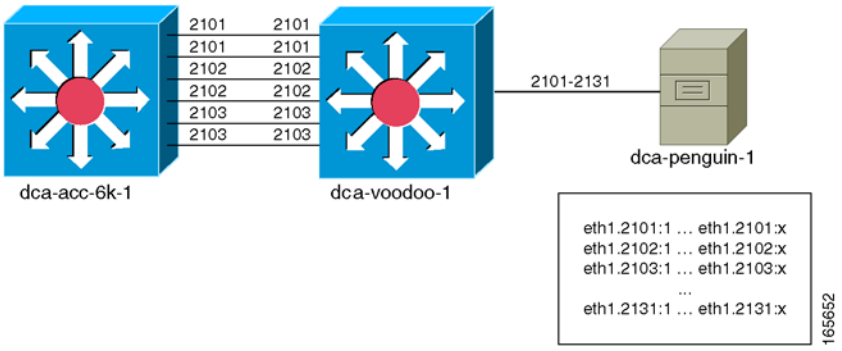


What Features are Used to Make Voodoo Work?

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.

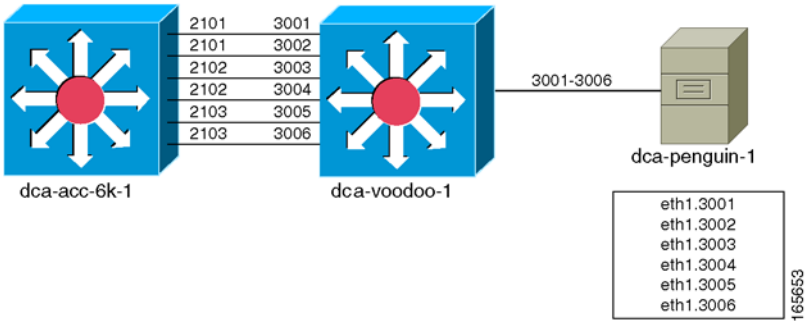
Figure H-2 Limitation of Voodoo with Virtual Interfaces



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.

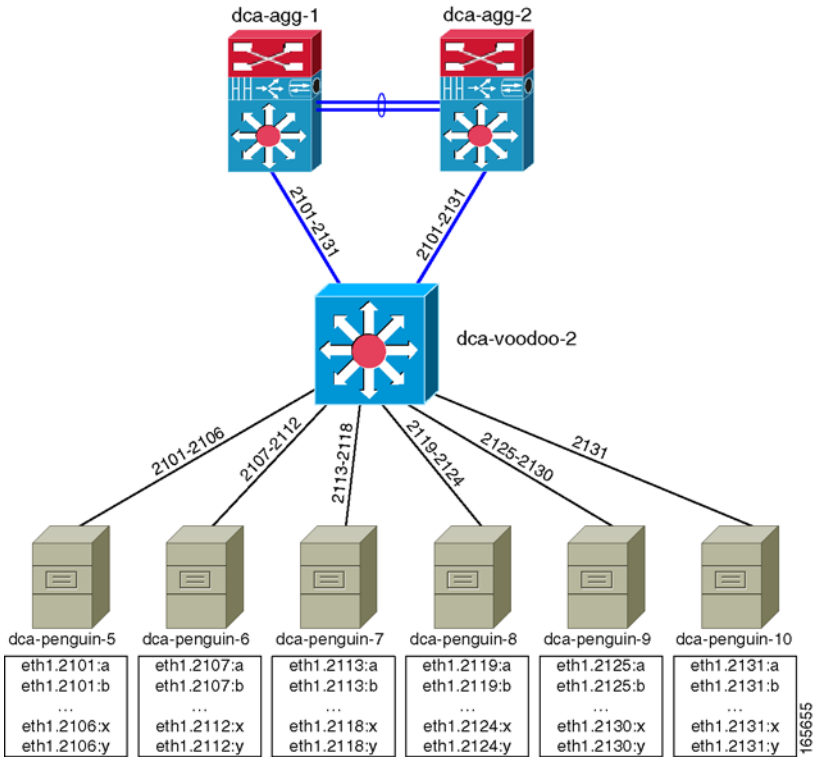
The new problem arises with the usage of the ARP protocol to resolve the MAC address of the default gateway for the emulated servers. It is a given that the 802.1q subinterfaces that share a similar IP subnet also share a common default gateway. So when the Linux box ARPs to resolve the gateway’s IP 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



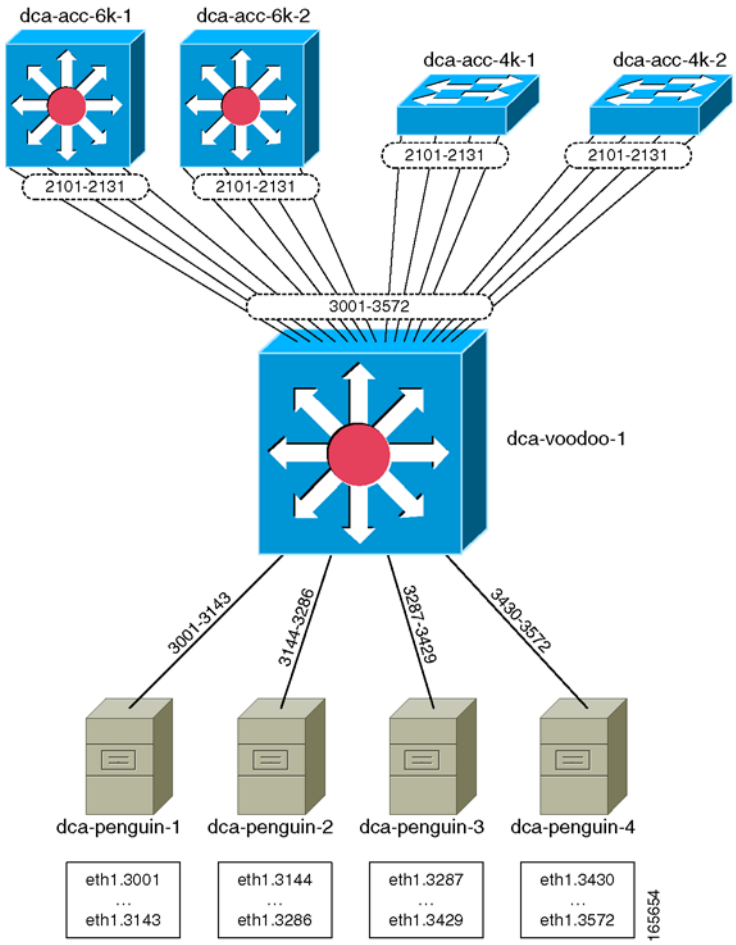
Again, a Catalyst 6513 fully populated with 48-port line cards is used as the Voodoo device, which yields 572 ports for physical links in the access layer. As will be noted below, each Linux server has a limitation on the number of source routes that can be configured (252), so at least 3 Linux servers 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 DCAP 3.0. It is broken down by device, and includes real power draw information taken with the device in idle state.

Table I-1 Cisco DCAP 3.0 Bill of Materials

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 3.0 Bill of Materials (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 WS-X6066-SLB-APC WS-SVC-SSL-1 (2) WS-SVC-NAM-2 WS-X6708-10GE (4) WS-X6748-GE-TX WS-F6700-DFC3CXL (2) WS-F6700-DFC3C (2)	2739
dca-agg-2	WS-C6513 WS-CAC-6000W WS-C6K-13SLT-FAN2 WS-SUP720-3B WS-SVC-FWM-1 WS-X6066-SLB-APC WS-SVC-SSL-1 (2) WS-SVC-NAM-2 WS-X6704-10GE (4) WS-X6748-GE-TX WS-F6700-DFC3B (4)	2520
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
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

Table I-1 Cisco DCAP 3.0 Bill of Materials (continued)

Device Name	Hardware List	Power Draw (watts)
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-F6700-DFC3BXL	696
dcb-core-2	WS-C6506-E WS-CAC-2500W (2) WS-C6506-E-FAN WS-SUP720-3BXL WS-X6704-10GE WS-F6700-DFC3BXL	696
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 3.0 Bill of Materials (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-NAM-2 WS-X6704-10GE	1252
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-NAM-2 WS-X6704-10GE WS-F6700-DFC3BXL	1252
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
dcb-acc-4k-2	WS-C4948-10GE	240

Table I-1 Cisco DCAP 3.0 Bill of Materials (continued)

Device Name	Hardware List	Power Draw (watts)
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 3.0 Bill of Materials (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 3.0 Bill of Materials (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-512-cm	WAE-512-K9	120
dcb-wae-7326-1	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-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 3.0 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 3.0 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

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 that the reader can locate and review the results of relevant Safe Harbor testing. A comprehensive list of the test cases executed in these other projects is provided in the Appendix to this document.

The results for SSLM 2.1(10) testing were used, along with undocumented testing on 2.1(11) to cover those areas potentially impacted by a single defect fix in 2.1(11).

Table K-1 Safe Harbor Technology Releases in EDCS

Platform	Software Version	EDCS Doc. No.
Supervisor 720	12.2(18)SXF7	583951
Firewall Services Module	2.3(3.2)	523606
Content Switching Module	4.2(6)	605556
Secure Socket Layer Module	2.1(11)	504160
Catalyst 4948-10GE	12.2(31)SGA	N/A
MDS9500	3.1(2)	N/A

The following summary tables list tests conducted in the latest Safe Harbor technology releases.

Native (Classic) IOS 12.2(18)SXF7

Table K-2 summarizes tests executed as part of the Cisco DCAP 3.0 initiative. Table K-2 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.

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-2 Safe Harbor Cisco DCAP 3.0 Testing Summary

Test Suites	Features/Functions	Tests
Housekeeping	Baseline	Network Steady State
		Network Validation
	System UpgradE	12.2(18)SXB11 to 12.2(18)SXF7—Sup 22
		12.2(18)SXB11 to 12.2(18)SXF7—Sup 720
		12.2(18)SXE5 to 12.2(18)SXF7—Sup 720
		12.2(18)SXF6 to 12.2(18)SXF7—Sup 22
		12.2(18)SXF6 to 12.2(18)SXF7—Sup 720
		Fast System 12.2(18)SXB11 to 12.2(18)SXF7—Sup 22
		Fast System 12.2(18)SXB11 to 12.2(18)SXF7—Sup 720
		Fast System 12.2(18)SXE5 to 12.2(18)SXF7—Sup 720
		Fast System 12.2(18)SXF6 to 12.2(18)SXF7—Sup 22
		Fast System 12.2(18)SXF6 to 12.2(18)SXF7—Sup 720
Memory Leak	Memory Leak	Remove and Restore—Sup22
		Remove and Restore—Sup720
		Repeated Telnet—Sup22
		Repeated Telnet—Sup720
		Repeated SSHv1—Sup22
		Repeated SSHv1—Sup720
		Repeated SSHv2—Sup22
		Repeated SSHv2—Sup720
		Multicast Flap Memory

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Hardware Integrity	Hardware Redundancy	Power Supply Failure 2500W—Sup 22
		Power Supply Failure 2500W—Sup 720
		Power Supply Failure 6000W—Sup 22
		Power Supply Failure 6000W—Sup 720
		Supervisor Hot Insert—Sup 22
		Supervisor Hot Insert—Sup 720
		Change Switch Modes Basic Function—Sup 22
		Change Switch Modes Basic Function—Sup 720
		Change Switch Modes RPR+ Failover—Sup 22
		Change Switch Modes RPR+ Failover—Sup 720
		Change Switch Modes SSO Failover—Sup 22
		Change Switch Modes SSO Failover—Sup 720
		Failover RPR+ —Sup 22
		Failover RPR+ —Sup 720
		Failover SSO Non Stop Forwarding—Sup 22
		Failover SSO Non Stop Forwarding—Sup 720
		Compact Mode Standby Sup Reset—Sup 22
		Compact Mode Standby Sup Reset—Sup 720
		Compact Mode RPR+ Failover—Sup 22
		Compact Mode RPR+ Failover—Sup 720
		Compact Mode SSO Failover—Sup 22
		Compact Mode SSO Failover—Sup 720
		Truncated Mode Standby Sup Reset—Sup 22
		Truncated Mode Standby Sup Reset—Sup 720
		Truncated Mode RPR+ Failover—Sup 22
		Truncated Mode RPR+ Failover—Sup 720
		Truncated Mode SSO Failover—Sup 22
		Truncated Mode SSO Failover—Sup 720
	Hardware Reliability	LC in Second Sup Slot Reset
		LC DFC Reset after Write Mem—Sup 720
		Power Cycle
		Rapid Link Flap

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Layer 2 Features	Unidirectional Link Detection-Aggressive Mode	UDLD on Layer 2 Link—Sup 22 UDLD on Layer 2 Link—Sup 720 UDLD on Layer 3 Link—Sup 22 UDLD on Layer 3 Link—Sup 720

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Layer 2 Features	Port Aggregation Protocol (Channeling)	Basic L2 Channeling Configuration—Sup 22
		Basic L2 Channeling Configuration—Sup 720
		Basic L3 Channeling Configuration—Sup 22
		Basic L3 Channeling Configuration—Sup 720
		Basic L2 LACP and Negative IOS to CatOS—Sup 22
		Basic L2 LACP and Negative IOS to CatOS—Sup 720
		Basic L2 PAgP and Negative IOS to CatOS—Sup 22
		Basic L2 PAgP and Negative IOS to CatOS—Sup 720
		L2 EtherChannel Failure/Recovery—Sup 22
		L2 EtherChannel Failure/Recovery—Sup 720
		L3 EtherChannel Failure/Recovery—Sup 22
		L3 EtherChannel Failure/Recovery—Sup 720
		L2 EtherChannel Load Balancing
		L3 EtherChannel Load Balancing
		Gigabit Ethernet Module Reset—Sup 22
		Gigabit Ethernet Module Reset—Sup 720
		L3 10-Gigabit Ethernet Module Reset—Sup 720
		L3 10-Gigabit Ethernet Load Balancing
		L3 10-Gigabit EtherChannel Fail/Recover
		L2 DEC Flooding Due to Periodic Purge—Sup 22
		L2 DEC Lost PI E Due To Timeout—Sup 22
		L2 DEC Lost PI E Due To Timeout—Sup 720
		L2 DEC Shut on DEC Port Unicast Flood—Sup 22
		L2 DEC Shut on DEC Port Unicast Flood—Sup 720
		L2 DEC Traffic No Flood to Linecards—Sup 22
		L2 DEC Traffic No Flood to Linecards—Sup 720
		L2 DEC MN Race Conditions—Sup 22
		L2 DEC MN Race Conditions—Sup 720
		L2 DEC to DEC Switching—Sup 22
		L2 DEC to DEC Switching—Sup 720
		L2 DEC Spanning Tree Mode Change—Sup 720
		L2 DEC MAC OOB Stress—Sup 720
		L2 DEC MAC OOB Sync Feature RPR+—Sup 720
		L2 DEC MAC OOB Sync Feature SSO—Sup 720

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Layer 2 Features	Trunking	Configuration and Failure Recovery—Sup 22
		Configuration and Failure Recovery—Sup 720
		VLAN Functionality—Sup 22
		VLAN Functionality—Sup 720
Layer 3 Routing Features	IP Multicast	Start Sources then Start Receivers
		Start Receivers then Start Sources
		First Hop Router Functionality—Sup 22
		First Hop Router Functionality—Sup 720
		Last-Hop Router Functionality—Sup 22
		Last-Hop Router Functionality—Sup 720
		Static RP Functionality—Sup 22
		Static RP Functionality—Sup 720
		Static RP Failover First Hop Router Impact—Sup 22
		Static RP Failover First Hop Router Impact—Sup 720
		Static RP Failover Impact—Sup 22
		Static RP Failover Impact—Sup 720
		Static RP Failover Traffic Impact—Sup 22
		Static RP Failover Traffic Impact—Sup 720
		Auto RP Functionality—Sup 720
		Auto RP Failover First Hop Router Impact—Sup 22
		Auto RP Failover First Hop Router Impact—Sup 720
		Auto RP Failover RP Impact—Sup 720
		Auto RP Failover Traffic Impact—Sup 720
		PIM DR Failover—Sup 22
		PIM DR Failover—Sup 720
		GEM Failover First/Last-Hop Router—Sup 22
		GEM Failover First/Last-Hop Router—Sup 720
		GEM Failover Layer 3 Interface—Sup 22
		GEM Failover Layer 3 Interface—Sup 720

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Layer 3 Routing Features	IP Multicast	IGMP Functionality—Sup 22
		IGMP Functionality—Sup 720
		IGMP Join/Prune Stress—Sup 22
		IGMP Join/Prune Stress—Sup 720
		IGMP Join/Leave Functionality—Sup 22
		IGMP Join/Leave Functionality—Sup 720
		MSDP SA Delay—22
		MSDP SA Delay—720
		Bidirectional PIM DF Election—Sup 720
		Bidirectional PIM RP Failover RP Impact—Sup 720
		Bidirectional PIM RP Failover Traffic Impact—Sup 720
		Layer 2 GEC Failover—SH3-110 to Dista-2
		Layer 2 GEC Failover—SH3-108 to Dista-1
		Layer 2 GEC Failover—SH3-106 to Dista-2
		Layer 2 GEC Failover—SH3-102 to Dista-1
		Layer 3 GEC Failover—SH3-104 to SH3-110
		Layer 3 GEC Failover—SH3-104 to SH3-109
		Layer 3 GEC Failover—SH3-104 to SH3-108
		Layer 3 GEC Failover—SH3-104 to SH3-107
		Layer 3 GEC Failover—SH3-100 to SH3-106
		Layer 3 GEC Failover—SH3-100 to SH3-105
		Layer 3 GEC Failover—SH3-100 to SH3-104
		Layer 3 GEC Failover—SH3-100 to SH3-102
		Layer 3 GEC Failover—SH3-100 to SH3-101
		Layer 3 GEC Failover—SH3-100 to SH3-97
		Multicast Stub and Non-RPF Rate Limiting —Sup 720
		GEM Failover on Auto RP—Sup 720
		10-GEM Failover on Auto RP—Sup 720
		PIM RPF Change Verification—Sup 720

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Layer 3 Routing Features	Border Gateway Protocol	BGP Functionality
		Route Scaling
		BGP Authentication Failure
		BGP Peer Flapping
		BGP Route Flap—With Dampening
		BGP Route Flap—No Dampening
		Inbound Route Map —Sup 720
		Outbound Route Map with Peer Group—Sup 720
		Redistribution OSPF to BGP
		Redistribution EIGRP to BGP
		Redistribution OSPF and EIGRP to BGP
	Cisco Express Forwarding	CEF Packet Switching—Sup 22
		CEF Packet Switching—Sup 720
		CEF FIB Consistency Verification—Sup 22
		CEF FIB Consistency Verification—Sup 720
		CEF Many-to-One Traffic Distribution
		CEF Many-to-Many Packet Distribution
	Enhanced Interior Gateway Routing Protocol	EIGRP Functionality
		EIGRP Authentication Failure
		EIGRP Neighbor Scaling
		EIGRP Route Summarization
		EIGRP Redistribution OSPF to EIGRP
	Hot Standby Routing Protocol	HSRP Failover with Default Timers—Sup 22
		HSRP Failover with Default Timers—Sup 720
		HSRP Failover with Fast Timers—Sup 22
		HSRP Failover with Fast Timers—Sup 720
		HSRP Traffic Impact on CPU—Sup 22
		HSRP Traffic Impact on CPU—Sup 720
		HSRP Recovery from System Failure—Sup 22
		HSRP Recovery from System Failure—Sup 720
		HSRP Maximum Group Limit—Sup 22
		HSRP Maximum Group Limit—Sup 720
		Distributed GEM Failover—Sup 720

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Layer 3 Routing Features	Open Shortest Path First	OSPF Basic Functionality OSPF Autocost OSPF Authentication Failure OSPF Passive Interface OSPF Filtering OSPF Redistribution EIGRP to OSPF OSPF Topology Database Verification
	Software Routing	Baseline Testing
Network Management Features	Simple Network Management Protocol	SNMP Functionality—Sup 22 SNMP Functionality—Sup 720 Config Copy via SNMP—Sup 22 Config Copy via SNMP—Sup 720 SNMP Malformed Packet Walk of DUT—Sup 22 Walk of DUT—Sup 720 Config Synch—Sup 22 Config Synch—Sup 720
	TACACS	User Authentication—Sup 22 User Authentication—Sup 720
Miscellaneous Features	Resiliency Verification	Steady-State Function and Resiliency Verification
	Security	802.1x Authentication with EAP/MD5—Sup 22 802.1x Authentication with EAP/MD5—Sup 720 802.1x Authentication Negative Tests—Sup 22 802.1x Authentication Negative Tests—Sup 720 NMAP Port Scanning—Sup 22 NMAP Port Scanning—Sup 720 Bad TACACS Login—Sup 22 Bad TACACS Login—Sup 720
	Crash TestinG	Software Crash with FTP Core File
	Web Cache Communication Protocol (WCCP)	WCCP—Version 1 Functionality WCCP—Version 2 Functionality
	Jumbo Frame	Jumbo Frame Support for Unicast Traffic
	Netflow Data Export	NDE Functionality—Sup 22 NDE Functionality—Sup 720

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Miscellaneous Features	Dynamic Host Control Protocol	DHCP Functionality—Sup 22
		DHCP Functionality—Sup 720
	Switched Port Analyzer	Handling of PIM Packets—Sup 22
		Handling of PIM Packets—Sup 720
		Transmit Only Multicast—Sup 22
		Transmit Only Multicast—Sup 720
		Receive Only Multicast—Sup 22
		Receive Only Multicast—Sup 720
		Transmit/Receive Multicast—Sup 22
		Transmit/Receive Multicast—Sup 720
		Transmit Only Unicast—Sup 22
		Transmit Only Unicast—Sup 720
		Receive Only Unicast—Sup 22
		Receive Only Unicast—Sup 720
		Transmit/Receive Unicast—Sup 22
		Transmit/Receive Unicast—Sup 720
		Remote Span Transmit Only Multicast—Sup 22
		Remote Span Transmit Only Multicast—Sup 720
		Remote Span Receive Only Multicast—Sup 22
		Remote Span Receive Only Multicast—Sup 720
		Remote Span Transmit/Receive Multicast—Sup 22
		Remote Span Transmit/Receive Multicast—Sup 720
		Remote Span Transmit Only Unicast—Sup 22
		Remote Span Transmit Only Unicast—Sup 720
		Remote Span Receive Only Unicast—Sup 22
		Remote Span Receive Only Unicast—Sup 720
		Remote Span Transmit/Receive Unicast—Sup 22
		Remote Span Transmit/Receive Unicast—Sup 720

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Miscellaneous Features	Access Control ListS	ACL 100—Sup 22
		ACL 100—Sup 720
		ACL 101—Sup 22
		ACL 101—Sup 720
		ACL 110—Sup 22
		ACL 110—Sup 720
		ACL 131—Sup 22
		ACL 131—Sup 720
	ParseR	Parser via Telnet—Sup 22
		Parser via Telnet—Sup 720
		Parser via Telnet BGP—Sup 720
		Parser via Telnet EIGRP—Sup 22
		Parser via Telnet EIGRP—Sup 720
		Parser via Telnet RIP—Sup 22
		Parser via Telnet RIP—Sup 720
		Parser via Telnet OSPF—Sup 22
		Parser via Telnet OSPF—Sup 720
		Parser via Telnet Show Platform—Sup 22
		Parser via Telnet Show Platform—Sup 720
		Parser via SSH—Sup 22
		Parser via SSH—Sup 720
		Parser via SSH BGP—Sup 720
		Parser via SSH EIGRP—Sup 22
		Parser via SSH EIGRP—Sup 720
		Parser via SSH RIP—Sup 22
		Parser via SSH RIP—Sup 720
		Parser via SSH OSPF—Sup 22
		Parser via SSH OSPF—Sup 720
		Parser via SSH Show Platform—Sup 22
		Parser via SSH Show Platform—Sup 720

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Miscellaneous Features	Network Address Translation	NAT Scaling—Sup 22
		NAT Scaling—Sup 720
		Static 2 Hosts—Sup 22
		Static 2 Hosts—Sup 720
		Static 2 Hosts with Multicast—Sup 22
		Static 2 Hosts with Multicast—Sup 720
		Static 2 Hosts with Jumbo Frames—Sup 22
		Static 2 Hosts with Jumbo Frames—Sup 720
		Static 2 Hosts with Jumbo UDP Frames—Sup 22
		Static 2 Hosts with Jumbo UDP Frames—Sup 720
		Static 40 Hosts—Sup 22
		Static 40 Hosts—Sup 720
		Static 40 Hosts Extended—Sup 22
		Static 40 Hosts Extended—Sup 720
		Static 40 Hosts Overnight—Sup 22
		Static 40 Hosts Overnight—Sup 720
		Static 2 Networks—Sup 22
		Static 2 Networks—Sup 720
		Static Inside Dynamic Outside—Sup 22
		Static Inside Dynamic Outside—Sup 720
		Dynamic Inside Static Outside—Sup 22
		Dynamic Inside Static Outside—Sup 720
		Addition/Removal of NAT Statements—Sup 22
		Addition/Removal of NAT Statements—Sup 720
		Increment Inside Random Outside Match Host—Sup 22
		Increment Inside Random Outside Match Host—Sup 720
	Network Time Protocol	NTP Functionality—Sup 22
		NTP Functionality—Sup 720
		NTP Failover—Sup 22
		NTP Failover—Sup 720
	Syslog	Syslog Functionality—Sup 22
		Syslog Functionality—Sup 720
	User Datagram Protocol Broadcast Flooding	UDP Broadcast Flooding—Sup 22
		UDP Broadcast Flooding—Sup 720

Table K-2 Safe Harbor Cisco DCAP 3.0 Testing Summary (continued)

Test Suites	Features/Functions	Tests
Miscellaneous Features	Secure Shell	SSH Server Vulnerability—Sup 22 SSH Server Vulnerability—Sup 720
	Integrated Data Center Quality of Service	IDC Basic QoS—Sup22 IDC Basic QoS—Sup 720 QoS Effects on OSPF Functionality—Sup 22 QoS Effects on OSPF Functionality—Sup 720 QoS Effects on HSRP Functionality—Sup 22 QoS Effects on HSRP Functionality—Sup 720 QoS Effects on STP Functionality—Sup 22 QoS Effects on STP Functionality—Sup 720 IDC Overnight Stress—Sup 22 IDC Overnight Stress—Sup 720 IDC Inband Ping—Sup 22 IDC Inband Ping—Sup 720

Firewall Services Module (FWSM) 2.3.3.2

The following Safe Harbor Firewall Services Module (FWSM) 2.3.3.2 release types were tested.

- Multi-Transparent Firewall Services Module (FWSM) 2.3.3.2, page K-14

Multi-Transparent Firewall Services Module (FWSM) 2.3.3.2

Table K-3 summarizes testing executed as part of the Cisco Safe Harbor Multi-Transparent Firewall Services Module (FWSM) 2.3.3.2 initiative. This table includes the technology tested, the feature or function tested, and the component tests for each feature or function.

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-3 Safe Harbor Multi-Transparent Firewall Services Module (FWSM) 2.3.3.2 Tests

Test Suites	Feature/Function	Tests
Baseline Testing	Baseline Testing	1. Short Lived HTTPS Flows 2. Long Lived FTP Flows 3. Multicast From Outside to Inside 4. Routing Updates
ACL for IP Traffic	ACL for IP Traffic	1. Object Group Auto Commit 2. Object Group Manual Commit 3. ACL Logging 4. Inbound ACL Manual Commit 5. 1K Inbound ACL with Traffic 6. 1K Outbound ACL with Traffic 7. 50K ACL Auto Commit Stress
Resource Usage	Resource Usage	1. Connection Rate Across Contexts 2. Syslog Rate Across Contexts 3. Fixup Rate Across Contexts 4. Total Connections Across Contexts 5. Total Xlate Across Contexts 6. Total SSH Session Across Contexts
Network Address Translation	Connection Limits—TCP and UDP Interception	1. Max. Connection Static NAT
	SYN Cookie	1. TCP SYN Attack 2. HTTP Performance Under SYN Attack 3. Passive FTP Under SYN Attack 4. Active FTP Under SYN Attack

Table K-3 Safe Harbor Multi-Transparent Firewall Services Module (FWSM) 2.3.3.2 Tests (continued)

Test Suites	Feature/Function	Tests
L7 Fixup	ICMP	1. ICMP Fixup 2. ICMP Error Fixup
L7 Fixup	FTP	1. FTP Fixup Passive FTP 2. FTP Fixup Active FTP
	DNS Fixup and DNS Guard	1. DNS Fixup Stress 2. DNS Fixup Max Reply Length 3. DNS Guard
Authentication, Authorization, and Accounting	AAA for Network Traffic	1. TACACS+ Authentication and Authorization 2. RADIUS Authentication and Authorization 3. Same User with Multiple IP 4. Multiple AAA server 5. AAA Authentication and Authorization Rate
	AAA for Admin Traffic	1. TACACS Authetication and Authorization for Mgmt Traffic 2. RADIUS AAA for Admin Traffic 3. TACACS+ Authentication and Authorization Fallback for Mgmt 4. RADIUS Authentication Fallback for Mgmt
	AAA HTTPS Proxy	1. HTTPS Cut-Through Proxy
	AAA Virtual Server	1. Virtual Telnet Server
Miscellaneous Features	Simple Network Management Protocol (SNMP)	1. SNMP Walks
	SYSLOG	1. Syslog Functionality 2. Syslog Performance 3. Syslog Standby
	Parser	1. Parser for Config Mode 2. Parser for Enable Mode
Hardware Redundancy	Failover/Fallback FWSM	1. Failover on Stateless Connection 2. Failover on Stateful Connection 3. Failover on Long Lasting Connection Traffic 4. Failover with Mcast Traffic 5. Failover with Config Sync Disabled 6. Failover with Suspend Config Sync and Manual Commit

Table K-3 Safe Harbor Multi-Transparent Firewall Services Module (FWSM) 2.3.3.2 Tests (continued)

Test Suites	Feature/Function	Tests
Hardware Redundancy	Failover/Fallback Switch	1. Fail Active FWSM
	Failover/Fallback Links	1. Fail Stateful Links 2. Fail Data Links 3. Fail Link to Upstream Router 4. Fail Link to Access Switch

Content Switching Module (CSM) 4.2.6

Table K-4 and Table K-5 summarizes testing executed as part of the Cisco Safe Harbor Router Mode Content Switching Module (CSM) 4.2.6 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-4 Safe Harbor Routed Mode Content Switching Module (CSM) 4.2.6 Tests

Features/Functions	Tests
CSM Basic Functionality	1. Additional Removal of Servers 2. CLI Parser 3. DEC rSPAN CSM Load Balance—Sup 720 4. DEC SPAN CSM Load Balance—Sup 720 5. Failaction Purge 6. Lifetime of Idle Connections 7. Route Health Injection 8. SNMP MIBs Traps 9. XML Negative Testing 10. XML
Health & Redundancy	1. Backup Serverfarm 2. Config Synch 3. Config Synch Large 4. Health Probes 5. Interface Tracking 6. Interswitch Redundancy 7. Intraswitch Redundancy 8. Module Reset 9. Port Reset 10. VLAN Reset

Table K-4 Safe Harbor Routed Mode Content Switching Module (CSM) 4.2.6 Tests (continued)

Features/Functions	Tests
Load Balancing Predictors	1. IP Address Hash 2. Least Connection 3. MaxConn 4. Predictor Round Robin 5. SASP Bindid 6. SASP Invalid Message Length 7. SASP Load Balancing 8. SASP Protocol Version 9. SASP Registration and Member State 10. SASP Scaling 11. SASP Weights 12. SASP Wrong Predictor 13. Server Weighting 14. URL Hash
Traffic Handling	1. Anomalous TCP 2. Cookies Insert 3. Cookies Maps 4. Cookies Sticky 5. FTP Active 6. FTP Passive 7. Header Insert 8. Header Maps 9. Header Sticky 10. Load Balance Non TCP 11. Netmask Sticky 12. Non Secure Routed Mode 13. Persistence Rebalance 14. Ping Handling 15. Policy Ordering 16. Redirect Policy 17. SSL Sticky 18. URL Lengths 19. URL Maps 20. VIP Dependency

Table K-5 Safe Harbor Bridged Mode Content Switching Module (CSM) 4.2.6 Tests

Features/Functions	Tests
CSM Basic Functionality	1. Additional Removal of Servers—Bridged 2. Failaction Purge—Bridged 3. Lifetime of Idle Connections—Bridged
Health & Redundancy	1. Health Probes—Bridged 2. Interface Tracking 3. Interswitch Redundancy 4. Intraswitch Redundancy 5. Port Reset—Bridged 6. VLAN Reset—Bridged
Traffic Handling	1. Load Balance Non TCP 2. Ping Handling—Bridged 3. Redirect Policy—Bridged 4. VIP Dependency—Bridges

Secure Socket Layer Module (SSLM) 2.1.10 & 3.1.1

Table K-6 and Table K-7 summarizes testing executed as part of the Secure Socket Layer Module (SSLM) 2.1.10, and 3.1.1 Safe Harbor initiative, respectively. These tables include component tests for each feature or function.

Note The results for SSLM 2.1(10) testing were used, along with undocumented testing on 2.1(11) to cover those areas potentially impacted by a single defect fix in 2.1(11).

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-6 Safe Harbor Secure Socket Layer Module (SSLM) 2.1.10 Tests

Features/Functions	Tests
Secure Socket Layer (SSL)	1. Upgrade 2. CLI Parser 3. Manual Certificate Signing Request 4. Certificate and Key Importation with PEM Paste 5. Graceful Certificate Rollover 6. URL Rewrite 7. SSL Client Proxy Services 8. Client Authentication 9. HTTP Header Insert Policy Client IP Port 10. HTTP Header Insert Policy Client Cert 11. HTTP Header Insert Custom Header 12. HTTP Header Insert Session 13. HTTP Header Insert Policy ALL 14. SSL Termination

Table K-7 Safe Harbor Secure Socket Layer Module (SSLM) 3.1.1 Tests

Features/Functions	Tests
Secure Socket Layer (SSL)	1. Upgrade 2. CLI Parser 3. Manual Certificate Signing Request 4. Certificate and Key Importation with PEM Paste 5. Graceful Certificate Rollover 6. URL Rewrite 7. SSL Client Proxy Services 8. Client Authentication 9. HTTP Header Insert Policy Client IP Port 10. HTTP Header Insert Policy Client Cert 11. HTTP Header Insert Custom Header 12. HTTP Header Insert Session 13. HTTP Header Insert Policy ALL 14. SSL Termination 15. SSLSM Configuration Virtualization 16. Export Ciphers 17. Protected Key Storage 18. SSL Session Auto Renegotiation 19. Maximum Connections