Cisco Videoscape Distribution Suite Transparent Caching Software Configuration Guide

Release 5.7.3

February 2016

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.

Text Part Number: OL-28016-06

The following information is for FCC compliance of Class A devices: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

The following information is for FCC compliance of Class B devices: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If the equipment causes interference to radio or television reception, which can be determined by turning the equipment off and on, users are encouraged to try to correct the interference by using one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Modifications to this product not authorized by Cisco could void the FCC approval and negate your authority to operate the product.

OL-28016-06 i
iii

Preface

Revised: February 2016, OL-28016-06

The *Cisco Videoscape Distribution Suite Transparent Caching Software Configuration Guide* introduces the Cisco VDS TC solution and provides information for working with the different VDS TC configurations.

This guide is divided into three parts:

- **Part 1–VDS TC Introduction:** Presents VDS TC, explains its components, and helps you understand the concepts that are required for using this guide.
- **Part 2–Integrated Appliance Configuration:** Provides hands-on information and guidance for working with VDS TC Integrated Appliance configurations:
  - **Working with VDS TC Management Tools:** Describes how to use the VDS TC configuration management tools, including the VDS TC command-line interface (CLI), file upload and download features, configuration files, and SNMP. This information is required to perform the tasks explained in this guide.
  - **Configuring VDS TC:** Describes how to perform Integrated Appliance system maintenance, system configuration, and network configuration, and outlines the P2P software functionality.
  - **Using CLI Commands:** Describes how to work with the Integrated Appliance configuration CLI.
  - **Monitoring VDS TC:** Describes how to monitor the Integrated Appliance configuration solution by viewing networking and cache statistics.
  - **CLI Reference:** Provides a list of the CLI commands and their definitions that are available for Integrated Appliance configurations.
- **Part 3–Cluster Configuration:** Provides hands-on information and guidance for working with VDS TC Cluster configurations, using either Cisco Unified Computing System™ (UCS) C-Series Rack Servers or B-Series Blade Servers.
  - **Working with VDS TC Management Tools (Cluster):** Describes how to use the Cluster configuration management tools, including the VDS TC CLI, file upload and download capabilities, the configuration files, and SNMP. This information is required to perform the tasks explained in this guide.
  - **Configuring VDS TC (Cluster):** Describes how to perform system maintenance, system configuration, and network configuration, and outlines the P2P software functionality.
  - **Using CLI Commands (Cluster):** Describes how to work with the Cluster configuration CLI.
  - **Monitoring VDS TC (Cluster):** Describes how to monitor the Cluster configuration by viewing networking and cache statistics.

– **CLI Reference (Cluster):** Provides a list of the CLI commands and their definitions that are available for Cluster configurations.

Acronyms

Table 1 describes the conventions used in the this guide.

Table 1 Acronyms

Acronym	Stands For
ACL	Access Control List
CIDR	Classless Inter-Domain Routing
CLI	Command Line Interface
CMDB	Cache Management Database
CPU	Central Processing Unit
HASH ID	A unique identifier for a file that is retrievable using peer-to-peer protocol.
IPMI	Intelligent Platform Management Interface
LAN	Local Area Network
NAS	Network Array Storage
P2P	Peer-to-Peer
P2PP	Peer-to-Peer Protocol
PBR	Policy Based Routing
QoE	Quality of Experience
SNMP	Simple Network Management Protocol
SOL	Serial Over LAN
TTL	Time To Live

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as an RSS feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service. Cisco currently supports RSS Version 2.0.

PART 1

VDS TC Introduction

Part 1 of the *Cisco Videoscape Distribution Suite Transparent Caching Software Configuration Guide* explains the components of a Cisco Videoscape Distribution Suite Transparent Caching (VDS TC) solution and helps you understand the concepts that are required for using this guide.

This part contains the following chapter:

Overview

Cisco Videoscape Distribution Suite Transparent Caching (VDS TC) is a carrier class caching and content acceleration product line that enables ISPs to meet the growing challenges of the network industry. Cisco technology reduces bandwidth and network infrastructure costs by caching popular content within the service provider network. It improves the user Quality of Experience (QoE) by delivering content to the subscriber at the fastest possible rate. The quality of the video experience defines the broadband service quality of the service provider which ultimately determines its competitive edge and success.

VDS TC integrates highly scalable caching software with high-performance Cisco Unified Computing System™ (UCS) C-Series Rack Servers and B-Series Blade Servers, Cisco switches, and SAN storage. The VDS TC solution, combined with other Videoscape products such as Cisco Videoscape Distribution Suite for Internet Streaming (VDS IS), provides a complete platform for optimizing managed and unmanaged content delivery.

VDS TC supports popular services and applications that are based on P2P and HTTP protocols. With P2P traffic and HTTP traffic comprising the majority of Internet traffic today, the solution intercepts P2P network traffic, monitors P2P activity, caches requests, and serves the requested P2P files from the cache instead of the WAN.

The ability of VDS TC to efficiently cache network traffic reduces bandwidth demands and minimizes infrastructure costs. The VDS TC P2P support includes BitTorrent, BitTorrent uTP, Ares, and e-Donkey. HTTP support includes video streaming services such as YouTube, Netflix, large video file downloads, and operating system and gaming updates.

VDS TC easily integrates into your network. VDS TC is a scalable architecture that grows with your network requirements. You can deploy VDS TC as a single cache engine server, which is referred to as an Integrated Appliance configuration, or you can deploy up to 16 cache engine servers and 5 storage enclosures, which is referred to as a Cluster configuration. You can also deploy multiple clusters to increase scalability. This architecture can scale to a multigigabit solution and can support small to very large ISPs. Single cluster systems simultaneously support P2P and HTTP traffic.

Solution Building Blocks

The VDS TC solution is built on three key elements that together provide the optimal solution for optimizing managed and unmanaged content delivery. These elements are carrier grade and provide redundancy at all levels:

- **Powerful L2/L3 switch:**

  - The switch is responsible for the redirection of cacheable traffic (P2P and HTTP) to the caching engine and is placed seamlessly in the existing network, connecting to router or DPI devices that redirect HTTP and or P2P traffic.
  - The configuration is based on forwarding HTTP and P2P requests to the caching engines using L4 policy routing to ensure that the cacheable traffic, and only the cacheable traffic, is forwarded to the caching engine.
  - The switch is introduced into the network as an L2 switch that is connected with Gigabit Ethernet or 10 Gigabit Ethernet links.
  - The switch distributes HTTP and P2P traffic to cache engines, tracking cache engines status and bouncing traffic back to the router in case the cache engines fail.
  - It is important to emphasize that the switch does not interact with the adjacent equipment beyond Layer 2. Therefore, there is no change to the source or destination IPs. In an Integrated Appliance configuration, there is no change in the next hop route of the routers on either side of the switch.

- **Caching engine:** The heart of the VDS TC caching solution is the caching engine. The caching engine analyzes P2P and HTTP traffic, and is in charge of the actual bandwidth savings that are achieved by caching and retrieving files from the centralized storage array. In addition, the Cluster configuration provides multiple I/O processing modules. Cisco UCS rack mount servers and blade systems provide these services.
  - In an Integrated Appliance configuration, the caching engine is provided by the Cisco UCS C240. In a Cluster configuration, the caching engine is based on modular technology that can scale to a multigigabit solution and uses either Cisco C220 Rack Servers or Cisco B200 Blade Servers.
- **Centralized storage array:** This is a high performance system that answers the extreme demands that are posed by cacheable traffic. In a VDS TC Cluster configuration, the centralized storage eliminates content duplication, which provides a scalable platform for increased bandwidth. A VDS TC Integrated Appliance configuration contains a set of internal disk drives that are used as the internal storage.

VDS TC is managed through an out-of-band network that is separate from the data flow, providing ultimate security.

Figure 1-1 illustrates a VDS TC Integrated Appliance configuration, Figure 1-2 illustrates a VDS TC C-Series Cluster configuration, and Figure 1-3 illustrates a VDS TC Blade Server configuration.

Figure 1-1 Integrated Appliance Configuration

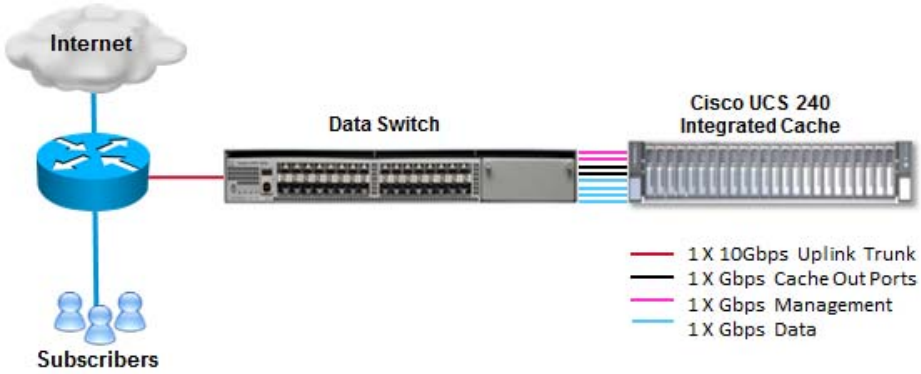


Figure 1-2 C-Series Cluster Configuration

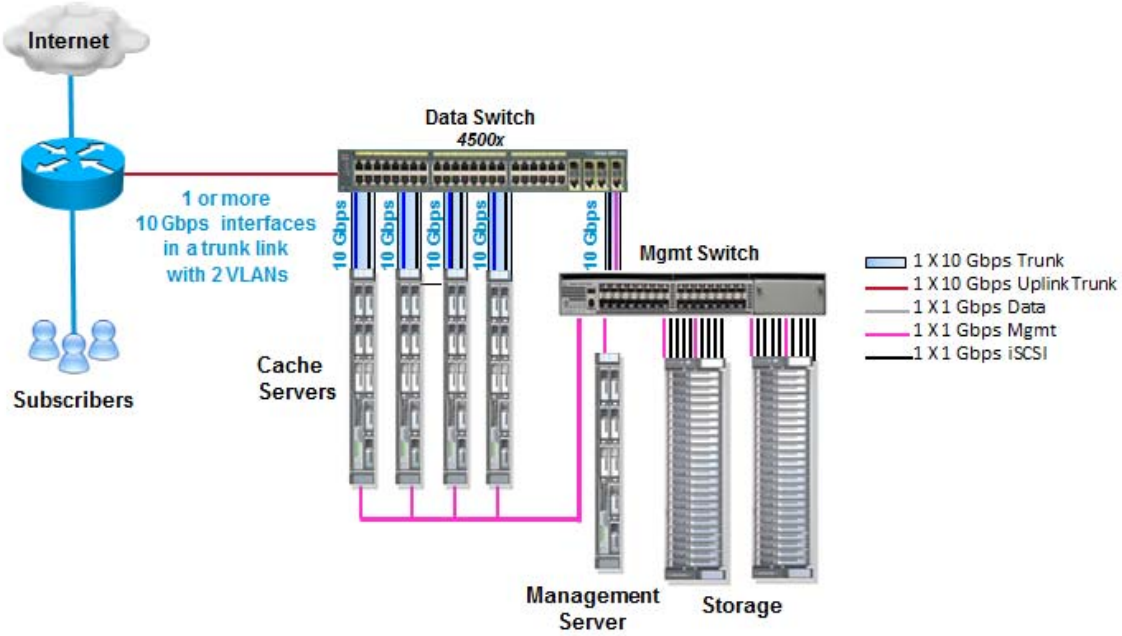


Figure 1-3 Blade Server Cluster Configuration with 1Gb Connections to the Storage Enclosure

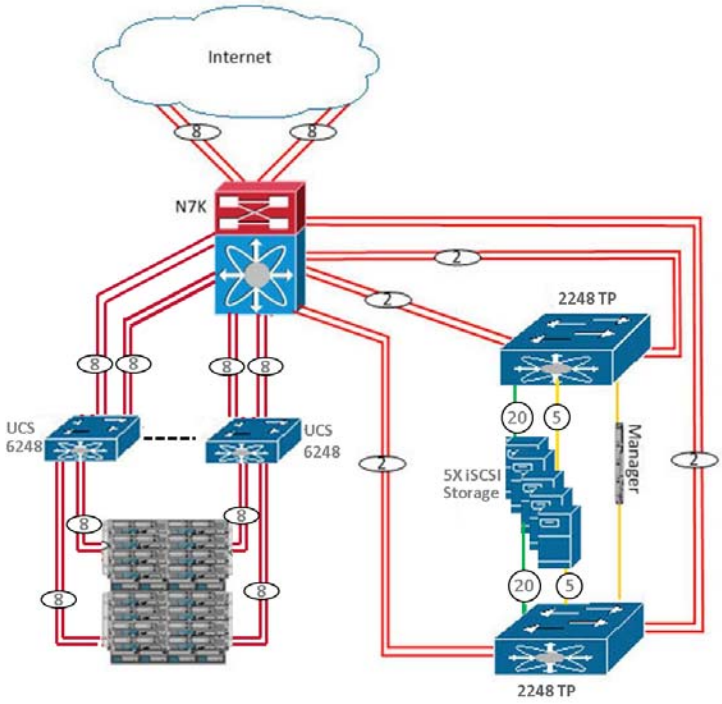
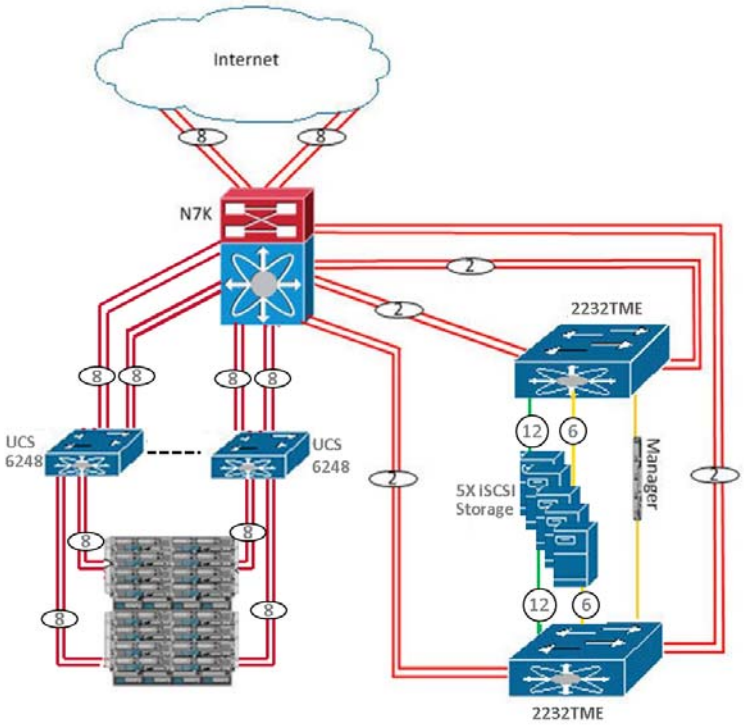


Figure 1-4 Blade Server Cluster Configuration with 10Gb Connections to the Storage Enclosure



Network Connectivity

Gigabit Ethernet links connect VDS TC externally to the network, either as single or multi mode fiber, or copper. In a Cluster configuration, internal links between the caching engines, management server, and storage arrays are Gigabit Ethernet copper links.

Figure 1-5 illustrates the VDS TC network connectivity for an Integrated Appliance configuration, Figure 1-6 illustrates the VDS TC network connectivity in a Cisco C-Series Cluster configuration, and Figure 1-7 illustrates the VDS TC network connectivity in a Cisco Blade Server Cluster configuration.

Figure 1-5 Network Connectivity: Integrated Appliance Configuration

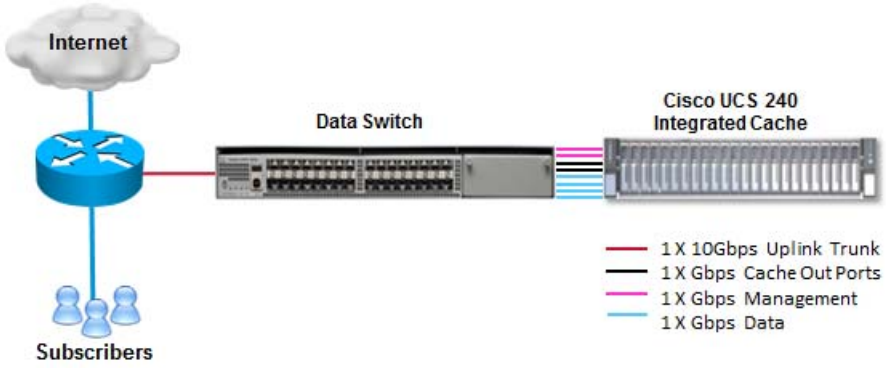


Figure 1-6 Network Connectivity: C-Series Cluster Configuration

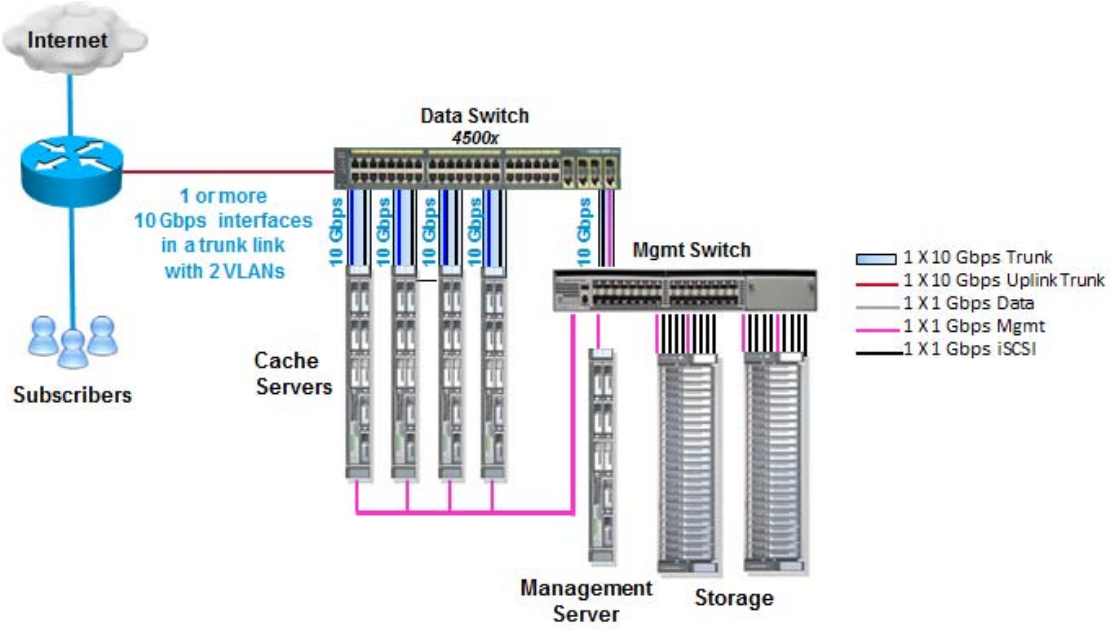


Figure 1-7 Network Connectivity: Blade Server Cluster Configuration with 1Gb Connections to the Storage Enclosure

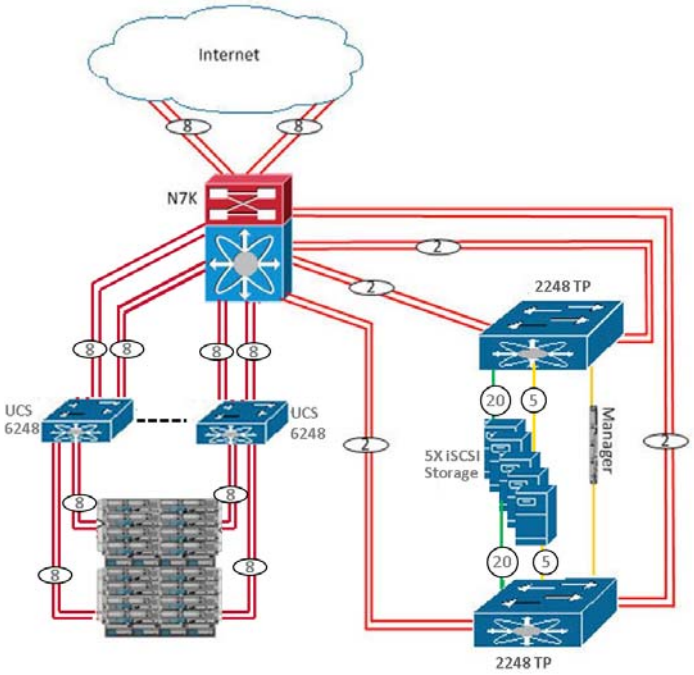
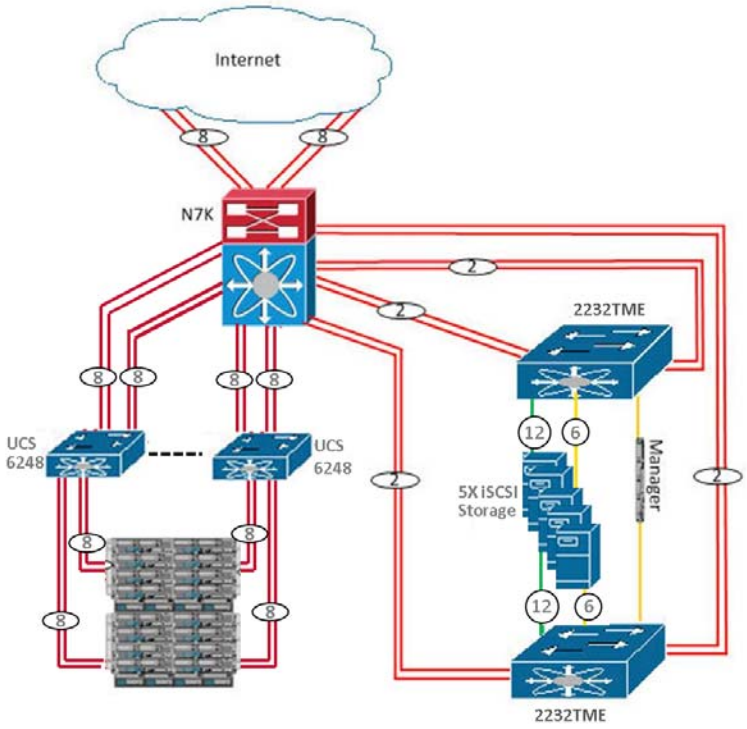


Figure 1-8 Blade Server Cluster Configuration with 10Gb Connections to the Storage Enclosure



Storage Connectivity Architecture

VDS TC can be configured to use one of the following types of storage devices, depending on the specific installation requirements:

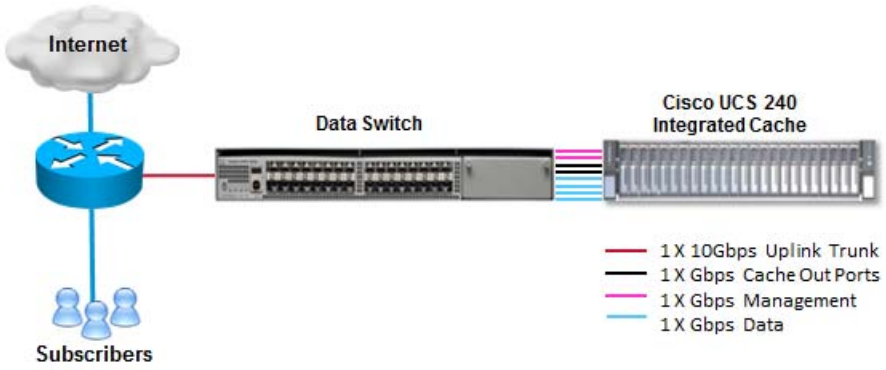
- **Integrated Storage:** The disks are hosted inside the Cisco UCS C240. This configuration is referred to as an Integrated Appliance configuration.
- **Storage Area Network (SAN):** NetApp E2724 or IBM DS3524 SAN solution. This solution is used for a VDS TC Cluster configuration.

Integrated Storage

The Integrated Appliance solution is equipped with integrated storage. In an Integrated Appliance configuration, the cache engine and the integrated storage devices share the same enclosure. This configuration was specifically created to address price performance requirements. The platform is equipped with two 300 GB SAS hard drives for the operating system and twelve 1 TB SATA hard drives for storage.

Figure 1-9 illustrates the VDS TC Integrated Appliance solution.

Figure 1-9 Hardware Topology - Integrated Storage Configuration



SAN Architecture

A VDS TC Cluster-ready configuration uses a NetApp E2724 SAN or an IBM DS3524 SAN. For a solution that uses the NetApp E2724, the storage devices are connected using a 10 Gb network adapter, via a dedicated VLAN on the VDS TC platform communication switch. For a solution that uses the IBM DS3524, the storage devices are connected using a 1 Gb network adapter, via a dedicated VLAN on the VDS TC platform communication switch. The NetApp E2724 and IBM DS3524 use the iSCSI protocol for communication and management.

Figure 1-10 illustrates the VDS TC SAN solution in a Cisco C-Series Cluster installation that uses a NetApp SAN and Figure 1-11 illustrates the VDS TC SAN solution in a Cisco C-Series Cluster installation that uses an IBM SAN.

Figure 1-12 illustrates the VDS TC SAN solution in a Cisco Blade Server Cluster installation that uses a NetApp SAN and Figure 1-13 illustrates VDS TC SAN solution in a Cisco Blade Server Cluster installation that uses an IBM SAN.

Figure 1-10 Hardware Topology - C-Series Cluster Configuration with NetApp SAN

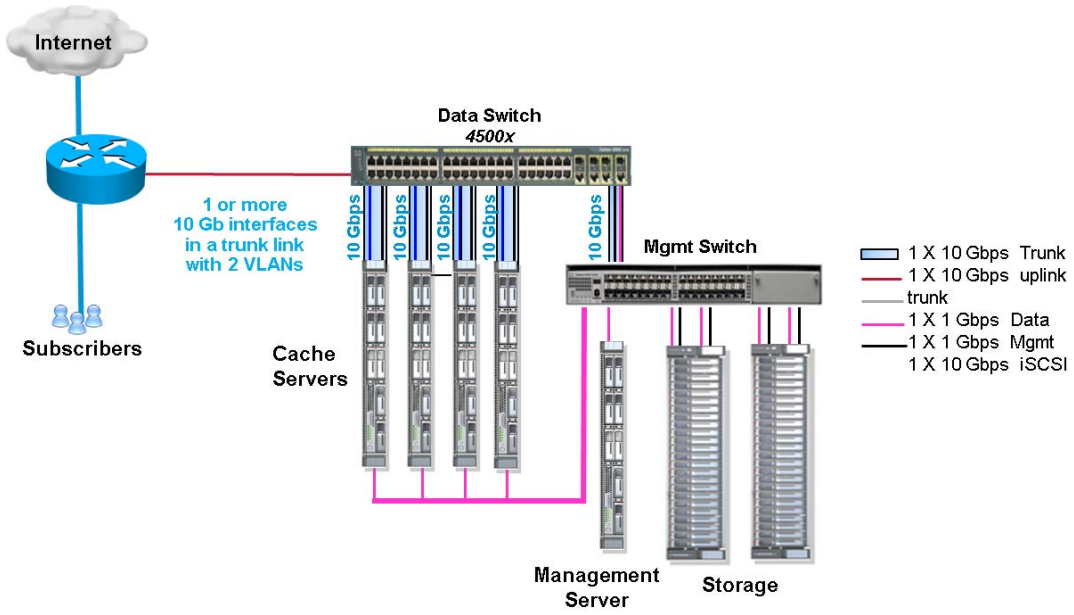


Figure 1-11 Hardware Topology - C-Series Cluster Configuration with IBM SAN

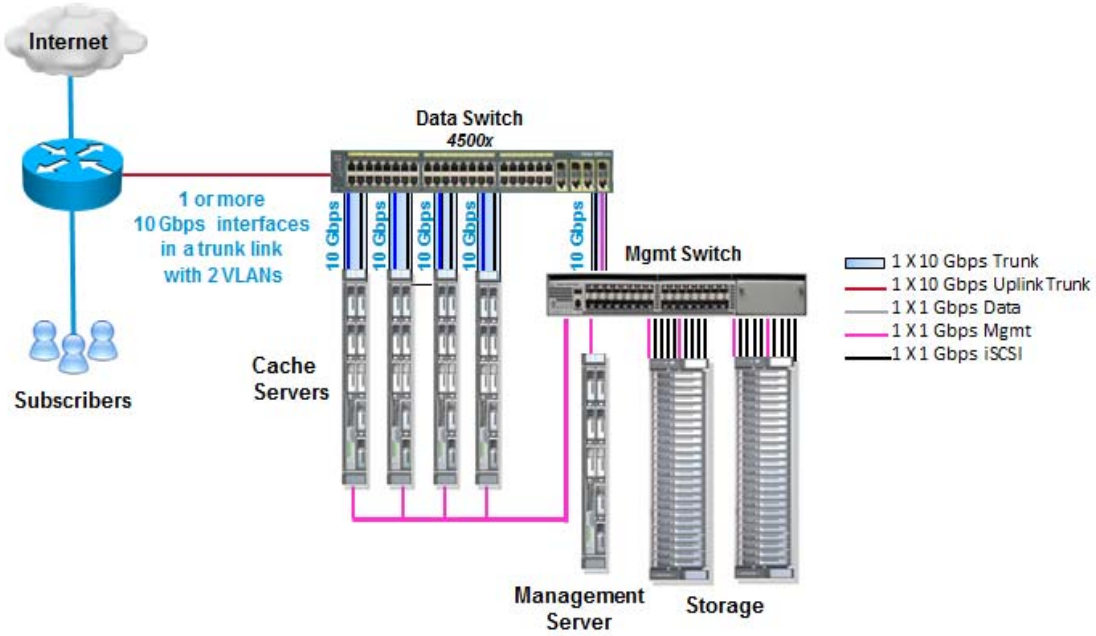


Figure 1-12 Hardware Topology - Blade Server Cluster Configuration with NetApp SAN

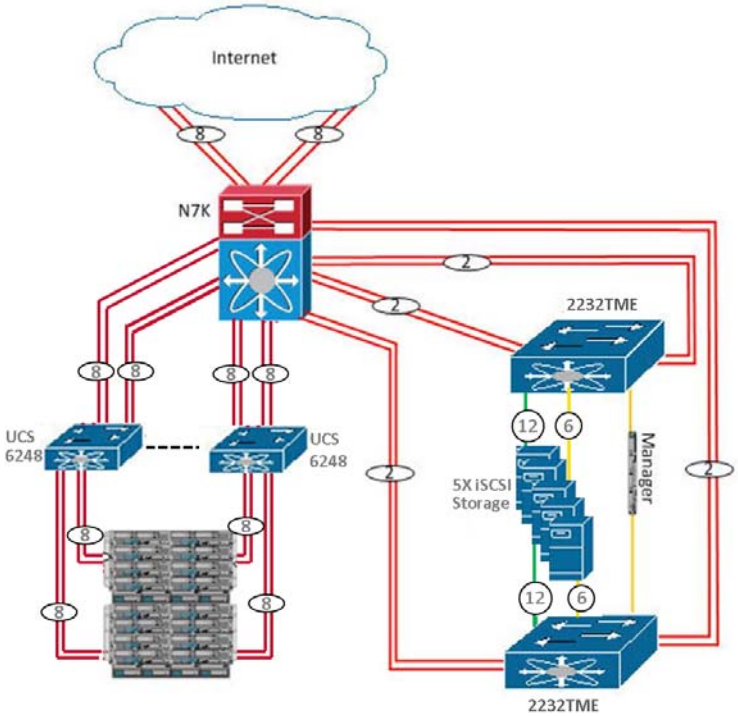
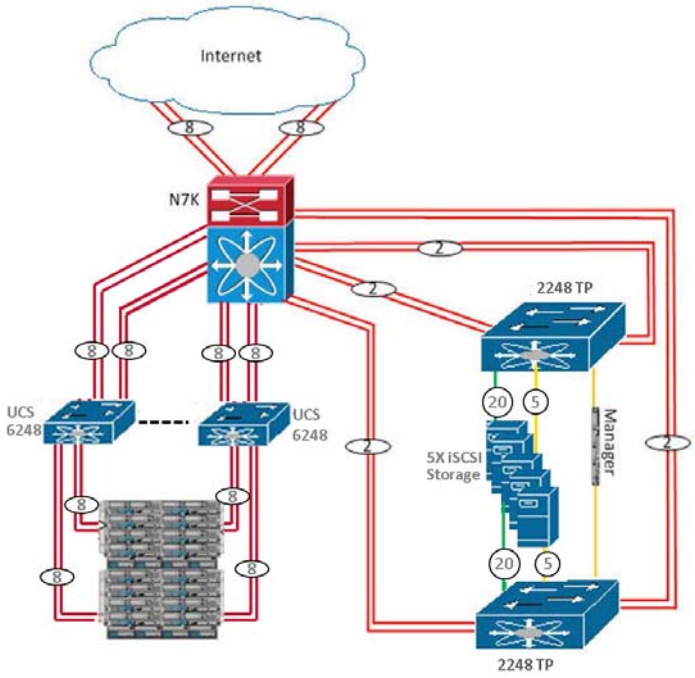


Figure 1-13 Hardware Topology - Blade Server Cluster Configuration with IBM SAN



Software Architecture

Cisco has developed high-performance caching engines that provide efficient manipulation of traffic. There are four distinct layers in the software architecture: networking, application, storage, and the distributed file system.

The network layer does not rely on the previous L4/L7 detection of P2P traffic, although previous P2P detection will result in improved performance.

The classifier, which is part of the networking layer of the software architecture, manages the TCP sessions and is capable of basic detection of P2P protocols. Non-P2P sessions, such as a miss in the MLS switch or DPI device, are forwarded transparently.

The inspection module, which is part of the application layer of the software architecture, works hand-in-hand with the classifier to define how to treat each session. This module is protocol aware. Once the session has been established, the classifier directs the session to the caching module. The caching module then either saves the data to disk and continues forwarding it to the client, or it intercepts the connection and serves the content from cache.

The storage layer is responsible for saving and retrieving data from the distributed file system. The disk manager decides what to save and maintains a local database. In a Cluster configuration, the local database is synchronized with the available content of all the other caching servers.

All four layers are accessible via a management API and are centrally controlled. In a Cluster configuration, the four layers are controlled by the management server. In an Integrated Appliance configuration, the four layers are controlled by the management software component running on the Cisco UCS C240.

Management

A VDS TC Cluster configuration utilizes a centralized management system. The management server is responsible for configuring, monitoring, and collecting data from all of the components within the VDS TC solution (switches, storage, I/O servers, chassis).

VDS TC Integrated Appliance and VDS TC Cluster configurations utilize a management component within the host. The management component is responsible for configuring, monitoring, and collecting data from the caching engine and the storage module.

For both Integrated Appliance and Cluster configurations, there are two ways to interact with the management system (centralized management system in Cluster configuration):

- **CLI:** The CLI is a context-based interface for easy configuration and basic monitoring. The CLI is a text-based interface that is accessible via a console cable and remotely using SSH (v2).
- **SNMP:** VDS TC provides a private MIB (SNMP v2) for easy access to all of the counters and information that are provided by VDS TC. This information is especially useful for customers who prefer using existing third-party software solutions. Monitoring using the private MIB offers centralized access to the VDS TC solution.

Configuration management also uses Intelligent Platform Management Interface (IPMI). IPMI provides autonomous monitoring and recovery features that are implemented directly in platform management hardware and firmware. The key characteristic of IPMI is that inventory, monitoring, logging, and recovery control functions are available independent of the main processors, BIOS, and operating system. Platform management functions can also be available when the system is in a powered down state.

IPMI capabilities are a key component to providing management for high-availability systems. IPMI enables you to obtain platform status information and initiate recovery actions in situations where system management software and normal in-band management mechanisms are unavailable.

Management Connectivity

There are two ways to physically connect to the management system for a VDS TC Cluster configuration and there are three ways to physically connect to the management system of a VDS TC Integrated Appliance configuration:

- **RS-232:** Console connection to the CLI, which is a text-based configuration and monitoring system.
- **Out-of-band Ethernet:** TCP connectivity to the management system. A cluster configuration also provides UDP connectivity to the management system. This connection provides remote access via SSH V2 (CLI) or SNMP.

  In an out-of-band management network, the network access control should allow access to the following ports for complete management functionality:

  - **TCP ports:** 22 (SSH), 161 (SNMP)
  - **UDP ports:** 161 (SNMP)
- **Serial Over LAN (SOL):** Remote console connection to the CLI that is redirected via an IPMI session over IP. Only an Integrated Appliance configuration provides this type of connection.

What a user can access is enabled using an access control list (ACL).

VDS TC provides two security levels for users, Regular and Enabled. The user database is stored locally on an Integrated Appliance.

- **Regular:** This security level allows you to perform read-only commands, monitoring, and basic debugging.
- **Enabled:** This security level allows you to perform configuration commands, maintenance procedures, and low level debugging.

System Configuration

Configurations are saved in text format and can be backed up on remote servers using TFTP. In an Integrated Appliance configuration, you can remotely manipulate the text-based configuration files and then download the files to the Integrated Appliance. In a Cluster configuration, you can remotely manipulate the text-based configuration files and then download them to the VDS TC management server, providing a means of easily maintaining and altering the configurations of multiple caching engines (this procedure is not recommended for either configuration.)

In an Integrated Appliance configuration, remote management enables the Cisco support team to remotely manage and monitor the service availability of the system:

- SNMP v2c support for industry-standard monitoring
- Intuitive SSHv2 CLI interface for system configuration using XML configuration files

System Monitoring

VDS TC provides the ability to remotely monitor the system, using the CLI or using a third-party monitoring system using SNMP and SNMP traps.

- **SNMP:** Provides access to all of the statistics and environmental information in real-time.
- **SNMP Traps:** Provides alarm triggered notification, including hardware failures, such as fans, temperature, power, and physical interfaces, and performance thresholds such as CPU utilization and bandwidth consumption.

Software Upgrades

VDS TC has two levels of software that are maintained; the operating system and the application.

In a Cluster configuration, operating system upgrades are infrequently required and usually contain security patches.

Typical Network Configurations

You can introduce VDS TC into a network using a centralized solution or locally per network zone, such as Metropolitan, POP, and cable.

Tip When performing a VDS TC network design, it is important to verify that the MLS switch or DPI device is placed on a link where both directions of the session pass through the device. This means that specific sessions always go through the same MLS switch or DPI device. If this is not the case, an asymmetric network topology setup is required, which will require two additional routers.

Table 1-1 describes the typical and supported ways of deploying VDS-TC.

Table 1-1 VDS TC Topology Solutions

Solution	Description
PBR/VRF routing solution	This topology is used when the MLS switch or DPI device is not inline and the router forwards at L4.
Basic inter-route solution	This topology is for an ISP that wishes to put the MLS switch or DPI device inline.
CMTS solution	This topology is used when a cable service provider wishes to save bandwidth within its network.
BRAS aggregation solution	This topology is the same as CMTS but traffic is further aggregated.
Asymmetric data solutions	These topologies may be used in cases where ingress and egress traffic may traverse different routes.

VDS TC is scalable to a multigigabit system and has been designed to grow with the network requirements.

- The basic Integrated Appliance configuration consists of an MLS switch or DPI device and a UCS C240 Integrated Cache and Manager Server. The capacity of this solution can be increased by adding additional VDS TC systems in parallel using etherchannel (port trunking) per site and increasing the storage capacity.
- The basic Cluster configuration consists of the following:
  - Data switch
  - Storage and management switch
  - A multi-I/O server chassis
  - 2 to 16 I/O servers, using either hardware servers or blade servers
  - A single management server
  - Multiple storage arrays

  Introducing this solution to a network provides redundancy of I/O data engines, management functionality, and the raid disk array. The capacity of this solution can be increased by mounting additional caching engines per chassis, multiple chassis, and increasing the storage capacity.

To supply redundancy for a MLS switch or DPI device, another MLS switch or DPI device should be deployed in a hot standby topology, much like that used in common networks.

The capacity of the solution can be increased by adding additional VDS TC systems in parallel using etherchannel (port trunking) per site and increasing the storage capacity.

PART 2

Integrated Appliance Configuration

Part 2 of the *Cisco Videoscape Distribution Suite Transparent Caching Software Configuration Guide* provides hands-on information and guidance for working with VDS TC Integrated Appliance configurations.

This part contains the following chapters:

Working with Cisco VDS TC Management Tools

This chapter describes how to use the management tools that come with Cisco Videoscape Distribution Suite Transparent Caching (VDS TC), including the CLI, file upload and download capabilities, the configuration file, and SNMP. The information provided in this chapter is required to perform the tasks explained in this guide.

VDS TC uses a number of different tools to help you configure, manage, and monitor its performance. Some management tasks allow you a choice of tool. For example, you can use both the CLI and SNMP to view statistics.

Table 2-1 lists the different VDS TC management tools that are explained in this chapter.

Note You can also use the VDS TC Manager to manage the VDS TC caching engine. For information on managing the VDS TC caching engine using VDS TC Manager, see *Cisco Videoscape Distribution Suite Transparent Caching Manager User Guide*.

Table 2-1 Management Tools

Management Tool	Description
CLI	Used to perform the bulk of configuration and management tasks. For a complete description of VDS TC CLI commands, *see* Chapter 4, "Using CLI Commands".
TFTP	Cisco provides license and software upgrades using upgrade files that are downloaded to the VDS TC installation using a TFTP server. In addition, you can modify the configuration by downloading a configuration file using TFTP.
Configuration File	Used to configure operational modes, caching, and SNMP settings.
SNMP	Used to monitor or view VDS TC operational statistics. Note: Information that you can view using SNMP is also available using the CLI.

Working with the CLI

You can use the CLI to perform configuration, management, and monitoring tasks, such as:

- Configure management settings, including:
  - Passwords
  - Management IP addresses
  - Local time
- Perform system maintenance, including:
  - Managing the caching service
- Monitor the system. You can also use SNMP to monitor the system.
- Upgrade the system, including:
  - Upgrade the software
  - Upgrade the license
  - Upgrading the system software or license is performed by downloading a new software or license file to the VDS TC device.

Note For a complete list of all CLI commands, *see* Chapter 4, "Using CLI Commands".

Getting Started with the CLI

There are two ways to access the VDS TC CLI:

- **Serial console:** The serial console is used to access the Regular CLI and the Rescue CLI. The Rescue CLI is used for entering the basic network and login information that you need to get the system up and running and is only available from the serial console.
  - To access the Regular CLI, log in as **admin**.
  - To access the Rescue CLI, log in as **rescue**.

  During the VDS TC installation process, the passwords for both the admin and rescue users are set to a specific system serial number. You should change these passwords after installation. For instructions on changing passwords, *see* Configuring Passwords, page 3-18.
- **LAN connection using SSH:** Use SSH to connect to the CLI over a LAN connection for regular access to the CLI. When accessing the CLI using a LAN connection, login using the username **admin**.

Note Before configuring the network settings for VDS TC, or if you have changed your network settings so that VDS TC is not accessible from the outside, you must use the serial console connection.

Follow these steps to log into the CLI using the serial console:

Step 1 Connect to the console port of the VDS TC host. In the terminal emulator communications software (such as HyperTerminal or Tera Term) configure the following settings:

- Speed: 57600
- Data bits: 8
- Parity: None
- Stop bit: 1

Step 2 Press **Enter** and a login window appears.

Step 3 Enter **rescue** to access the Rescue mode and press **Enter**.

Step 4 Enter the user password and press **Enter**. Wait while the setup script completes and the system automatically logs out.

Follow these steps to log into the CLI using SSH over a LAN connection:

Step 1 Connect to VDS TC using SSH from anywhere on your LAN. A login prompt appears.

Step 2 Enter **admin** and press **Enter**.

Step 3 Enter the password for the admin user and press **Enter**. The CLI prompt `console>` appears.

Once you are logged into the CLI using either the serial console or LAN connection, enter **help** or press **?** to view a list of commands.

To logout of the CLI when you are done, enter **exit**.

CLI Command Editing Features

You can edit CLI commands using the following keystrokes:

- Press the **TAB** key to auto-complete a command. If multiple choices are available, they are displayed, one option per line.
- Press **?** to display a command and its parameter hints. Enter as much of the command as you know and then press **?** to display the completion options and their descriptions. Each option is displayed one per line.
- After the command completes, you can display the next parameter hint by pressing **SPACEBAR ?**.
- Use the Up arrow and the Down arrow keys to navigate through the command history.

CLI Modes

VDS TC supports the following CLI modes:

- **Regular mode:** From this mode you can view system configuration and statistics, but you cannot change the settings. In Regular mode, the console> prompt appears and you can either enter Enable mode or exit the CLI.
- **Enable mode:** From this mode you can update the license or software, set the date, and configure the login name and password. In Enable mode, the console# prompt appears and you can enter configuration mode or return to Regular mode using the **exit** command.
- **Configuration mode:** From this mode you can configure any settings on the system. In Configuration mode the configuration# prompt appears. You can return to Enable mode using the **exit** command.

Note Configuration mode can only be used by one user at a time.

Caution If you exit Configuration mode without applying your changes, these changes are lost.

- **Rescue CLI:** The Rescue CLI allows you to recover incorrect management network configurations and lost or forgotten CLI passwords. In Rescue CLI mode the rescue@ce-1# prompt appears and you can execute the following commands:
  - **access:** Resets the white and black management access lists.
  - **passwords:** Resets the admin and rescue passwords.
  - **network:** Configures the following network parameters: IP address, subnet mask, default gateway, and DNS server.
  - **exit:** Exits the Rescue CLI mode.
  - **help:** Displays a list of Rescue CLI commands. You can also display a list of Rescue CLI commands by pressing ? at the CLI rescue prompt.

  After executing these commands, you can immediately perform another command or you can exit the Rescue CLI mode.

Switching from Regular Mode to Enable Mode

Follow these steps to switch from Regular mode to Enable mode:

Step 1 At the CLI Regular mode prompt, enter **enable**. A password prompt appears.

Step 2 Enter the Enable mode password and press **Enter**. You are now logged into Enable mode. The prompt should now show console#.

Note Your system serial number is the default password for the Enable mode. Ensure that you change it at installation. For instructions on how to change passwords, see Configuring Passwords, page 3-18.

While logged into Enable mode you may need to run configuration commands. To run configuration commands you must enter Configuration mode. To enter Configuration mode from the Enable mode, enter **config**. You will now be in Configuration mode.

To exit Configuration mode, enter **exit**. You are returned to Enable mode. If you enter **exit** in Enable mode, you are returned to Regular mode.

TFTP Server

You use a TFTP server to upload files to and download files from VDS TC. You download files to VDS TC to do the following:

- Update the VDS TC software
- Update the VDS TC license
- Update the configuration by changing the configuration file

Before downloading a file to VDS TC, you must place the file on the TFTP server. You can use an external TFTP server or you can use the TFTP service that runs on the VDS TC server. If you use the TFTP service that runs on the VDS TC server, use **localhost** for the TFTP server name in the CLI commands. When using localhost as the TFTP server, the files will be uploaded to or downloaded from the /tftpboot directory.

Note You can also upload files to VDS TC using VDS TC Manager. This eliminates the need to work with an external TFTP server.

Working with the Configuration Files

There are three configuration files that are used to configure different aspects of the VDS TC platform. Each file is responsible for different operational activities:

- The main configuration file is an XML file that manages operational modes, cache settings, and SNMP settings for VDS TC. The common name for this file is cluster_conf.xml.
- Two additional configuration files define the traffic categorization rules: one file contains the major categories, also referred to as groups, and the other file contains the subcategories, also referred to as signatures. For example, a major category listed in the groups file might be Video Streaming, and its subcategories listed in the signatures file might be youtube, google.video, and video.facebook. These files are also XML files.

To modify the functionality of the VDS TC system, you can make changes directly to these configuration files and then upload the changes to the system.

Changing the Configuration Files

To make changes to the configuration files and apply these changes to the VDS TC system, follow these steps:

	Command	Purpose
Step 1	`console# **config**`	Enters Configuration mode.
Step 2	`configuration# **export** *TFTP_server filename*`	Exports the current configuration file to a TFTP server. • The *TFTP_server* parameter is the IP address or hostname of the TFTP server to which you want to export the configuration file. To export the file to the /tftpboot directory on the VDS TC platform use **localhost** for this parameter. • The *filename* parameter is the name of the file to which you want to save the current configuration. The common filename to use is cluster_conf.xml.
Step 3	`configuration# **exit**`	Exits Configuration mode and returns to Enable mode.
Step 4	`console# **detection_rules export_groups** *TFTP_server filename*`	Exports the major categories (groups) to a TFTP server. • The *TFTP_server* parameter is the IP address or hostname of the TFTP server to which you want to export the groups file. To export the file to the /tftpboot directory on the VDS TC platform use **localhost** for this parameter. • The *filename* parameter is the name of the file to which you want to save the major categories.

	Command	Purpose
Step 5	console# **detection_rules export_signatures** *TFTP_server filename*	Exports the subcategories (signatures) to a TFTP server. • The *TFTP_server* parameter is the IP address or hostname of the TFTP server to which you want to export the signatures file. To export the file to the /tftpboot directory on the VDS TC platform use **localhost** for this parameter. • The *filename* parameter is the name of the file to which you want to save the subcategories.
Step 6	Open the configuration file in a text editor or an XML editor.	Make the desired changes to the configuration file and save the changes. **Note** When editing the configuration file, edit only the field values and do not change or erase the XML markup tags. If XML tags are changed, the configuration will be rejected upon loading.
Step 7	console# **config**	Enters Configuration mode.
Step 8	configuration# **import** *TFTP_server filename*	Imports the new main configuration file to VDS TC. • The *TFTP_server* parameter is the IP address or hostname of the TFTP server from which you are importing the configuration file. To import the file from the /tftpboot directory on the VDS TC platform use **localhost** for this parameter. • The *filename* parameter is the name of the configuration file that you want to import.
Step 9	configuration# **display**	Displays the currently loaded configuration. This configuration will not take effect until you enter the **apply** command. To view the differences between the currently active configuration and the newly imported configuration, use the **diff** command. The differences are marked with a plus (+) and a minus (-): • **minus (-):** Represents the exiting configuration, prior to the import. • **plus (+):** Represents the new configuration, after the import.
Step 10	configuration# **apply**	Applies the new configuration.

	Command	Purpose
Step 11	`configuration# display`	Displays the current configuration.
Step 12	`configuration# diff`	Shows pending changes. The differences between the currently active configuration and the newly imported configuration are marked with a plus (+) and a minus (-): • **minus (-):** Represents the exiting configuration, prior to the import. • **plus (+):** Represents the new configuration, after the import. If you have successfully applied the new configuration, this command should display "Configurations are identical".

Note To discard the pending changes, use the **discard** command in Configuration mode. If you have applied the changes and want to revert back to the last known good configuration, use the **restore** command in Configuration mode.

Configuration File Sections

The main configuration file has three main sections:

- **mgmt-config:** Use this section to configure the network settings on the system, such as the management IP address and default gateway.
- **common:** Use this section to define the default settings for the host.
- **blade id=#:** # represents the number of the caching engine blade. Only VDS TC Cluster configurations will have numbers higher than 1. Use the <blade id> section to define the settings for the caching service. Some of the fields in the <service> subsection of the <blade id> section are also in the <service> subsection of the <common> section. When there is a value for a field in both sections, the value of the field in the <blade id> section takes priority over the value of the field in the <common> section.

Accessing VDS TC SNMP Information

VDS TC provides a robust set of SNMP status information that you can monitor using any standard SNMP tool. All status information that is available using SNMP is also available using status commands at the CLI. The SNMP information is provided using a private MIB (SNMP v2) environment. The private MIB file is located in the /opt/pang/mgmt/avalon/share/snmp/mibs folder on the VDS TC Cache Engine and is named VDS-TC-MIB.txt.

Configuring VDS TC

This chapter describes the steps necessary to configure and perform maintenance on the VDS TC system if changes are required on a running platform. This chapter has the following sections:

Main Operational Features (Quick Jumpstart)

You must initially configure the following elements to jumpstart the system. Detailed configuration instructions are provided later in this chapter.

- CLI-based configuration:
  - Configure the management network
- File-based configuration:
  - Configure SNMP
  - Configure P2P protocols
  - Configure traffic forwarding options: When you deploy the VDS TC Integrated Appliance solution, there are several different supported L7 device configurations that can be used. Symmetric/asymmetric connection modes with single/multiple port connections are possible when you are configuring L7 devices with a VDS TC Integrated Appliance platform. The following are the supported configurations:
    - Improve connectivity options, by adding support to work with more than one physical Ethernet interface.

VDS TC Features

The following is a description of system features that are available in VDS TC. You can configure some of these features using the VDS TC configuration file and some of them using the VDS TC CLI.

Note Any changes that you make to the VDS TC configuration file will only take affect after you import the configuration file into the VDS TC management server. For information on how to import this file, see Working with the Configuration Files in Chapter 2, "Working with Cisco VDS TC Management Tools".

Caching Specific Features

- **Black list of hashes:** You can blacklist specific hashes as non-cacheable which prevents the hashes from being cached or provided again to users, if already cached. The black list is maintained using the CLI, allowing administrators to add, remove, or view the list of hash IDs in the black list. *See* the cache command, page 4-18.
- **Selective caching:** The VDS-TC platform includes a dynamic mechanism that automatically decides if a specific large content item should be cached-in as popular content. Based on this decision, made for each content item that is requested by a user, the content is either cached-in or forwarded to the user. The platform allows the operator to disable this mechanism. If you disable this feature, all content items that are requested by a user are immediately cached-in and ignore any "popularity-algorithm" decision making.

  Setting the <selective_cache_in_threshhold> parameter to 0 disables the selective caching feature and forces the caching-in of all large content items, ignoring the popularity that is associated with these files. To disable the selective caching feature enter the following configuration in the <policy> subsection of the <service> section:

  ```
  <selective_cache_in_threshhold>0</selective_cache_in_threshhold>
  ```

  Note Disabling the selective caching feature on a production platform will lead to caching all content items that are requested by users. This is not recommended because the storage will fill up rapidly and the cache-out performance will be dramatically decreased.

  Note

- **CIDR:** This feature is a method to allocate IP addresses and routing IP packets. This list of IP subnets defines the subscribers that will be served from the cache. You can change this feature in the configuration file and then import and apply the changes. The VDS TC caching service has a limit of 512 entries, so if there are many /24 subnets, you should aggregate them. The following is an example of this information in the configuration file in the <net> subsection of the <common> or <blade id="1"> section:

  ```
  <subnet_range_per_link name="a">
      <cidr_range>10.188.0.0/18</cidr_range>
      <cidr_range>192.168.32.0/20</cidr_range>
      <cidr_range>10.207.32.0/24</cidr_range>
      <cidr_range>10.207.33.0/24</cidr_range>
      <cidr_range>10.207.36.0/24</cidr_range>
      <cidr_range>10.207.37.0/24</cidr_range>
      <cidr_range>10.207.42.0/24</cidr_range>
      <cidr_range>10.207.43.0/24</cidr_range>
  ```

```
    <cidr_range>10.207.44.0/24</cidr_range>
    <cidr_range>172.16.175.250</cidr_range>
    <cidr_range>192.168.15.0/24</cidr_range>
</subnet_range_per_link>
```

- **Small memory buffer:** When traffic is very low because of a shaper configuration, the memory buffer size might be too large (in memory) for this shaper. This feature allows you to control the buffer size, to optimize memory use and tailor it to the way the traffic is shaped. If the buffer size does not match the shaped traffic size, the cache will fill up too slowly. You can configure the number of buffers in the configuration file.

  To control the number of buffers that are used, add or edit the following text in the <service> section in the configuration file where the *number* parameter is the number of buffers that you would like to use, such as 8000:

```
<memory>
        <small_io_blocks>number</small_io_blocks>
</memory>
```

- **Bandwidth-per-connection management:** This configuration option controls the cache-out sessions (bandwidth management), which enables you to place a top limit on the cache-out sessions.

  To modify the cache-out sessions, add or edit the following text in the <service> section of the configuration file, where the *max_bw_per_IP* parameter is the maximum bandwidth per IP in b/s:

```
<bandwidth-management>
    <enable-bandwidth-management>1<enable-bandwidth-management>
    <bandwidth-per-connection>max_bw_per_IP<bandwidth-per-connection>
</bandwidth-management>
```

- **Administrative state locked:** This configuration option prevents the server on which it is configured from handling traffic. The traffic is not processed.

  To lock the server, add the following text to the configuration file in the <blade id=#> section:

```
<cache-engine>
    <administrative_state>locked</administrative_state>
</cache-engine>
```

- **Upstream caching:** A last mile architecture suffers from limited upstream resources that are significantly affected by peer-to-peer symmetrical traffic pattern. Upstream caching relieves network congestion by providing cached pieces to peers in other “zones”, directly from the cache instead of the last mile user. There is no configuration for this feature.
- **HTTP caching:** You can configure VDS TC to cache HTTP documents (such as video files, video streaming, and images) to reduce bandwidth usage and to improve the user experience by providing accelerated document download time. When you implement transparent HTTP caching, any standard HTTP contained document can be cached regardless of the URL that is associated with it.

  To enable VDS TC to support the HTTP protocol, add the following text to the configuration file in the <protocols> subsection of the <service> section:

```
<enable-http>1</enable-http>
```

- **HTTP minimum file size:** You can modify the minimum size a file needs to be in order to be cached by the system. Files requested using HTTP that are larger than the value defined with the http_min_file_size parameter are cached by the system, if popular. Files that are smaller than this value are not cached by the system.

  The default value for the http_min_file_size parameter is 512 Kb. To change this value, add the following text in the <policy> subsection of the <service> section in the configuration file, where the *file_size* parameter is the minimum file size in bytes:

```
<http_min_file_size>file size</http_min_file_size>
```

For example:

```
<http_min_file_size>65536</http_min_file_size>
```

Note It is recommended that you only change the HTTP minimum file size at the default of 512 Kb if instructed to by your Cisco Support System Engineer.

- **Selective HTTP caching:** The VDS-TC platform includes a dynamic mechanism that automatically decides if a specific large HTTP item should be cached-in as popular content. Based on this decision, made for each content item that is requested by a user, the content is either cached-in or forwarded to the user. The platform allows the operator to disable this mechanism. If you disable this feature, all content items that are requested by a user are immediately cached-in and ignore any "popularity-algorithm" decision making.

Note Disabling this dynamic caching selection method is only appropriate in lab environments.

To disable the dynamic caching selection method so that the system starts caching the HTTP requests on the first request, add the following text in the <policy> subsection of the <service> section in the configuration file.

```
<http_selective_cache_in_threshhold>0</http_selective_cache_in_threshhold>
```

- **Flush hours:** The max_hours_hash_not_touched parameter sets the amount of time to wait, in hours, before flushing an empty HASHID (which is created when seeing a request for the file for the first time) from the cache index if the file is empty. The default value is 24 hours.

Note This parameter is for HTTP only.

To change this value, add or edit the following text in the <policy> subsection of the <service> section in the configuration file:

```
<max_hours_hash_not_touched>hours</max_hours_hash_not_touched>
```

For example:

```
<max_hours_hash_not_touched>48</max_hours_hash_not_touched>
```

- **Ares protocol support:** Ares Galaxy is an open source P2P file sharing application and protocol that uses its own decentralized supernode/leaf network.

  To enable the VDS TC system to support the Ares protocol, add the following text to the configuration file in the <protocols> subsection of the <service> section:

```
<enable-ares>1</enable-ares>
```

- **Do not cache specific URLs, hosts, or subnets:** You can configure VDS TC to avoid caching specific URLs, hosts, or subnets. Each URL seen in a GET request that floats through the VDS TC platform is compared to all of the URLs, hosts, and subnet entries in the <no_cache_url_list> section. When a match is found, the cache-in and cache-out are skipped and the request is forwarded as is. The configuration to avoid caching specific URLs, hosts, or subnets is configured in the <policy> subsection of the <service> section in the configuration file.

Note The <no_cache_url_list> section is limited to 64 entries. Use this option with caution. It imposes an extra burden on the VDS TC system because it compares all of the entries in the table with each URL floating through the system.

The following is an example of how to add a URL to the no cache list.

```
<policy>
<no_cache_url_list>
    <url_no_cache>video_id</url_no_cache>
    <url_no_cache>videoplayback</url_no_cache>
</no_cache_url_list>
</policy>
```

This example will match www.thegame.com/video_id/movie=8979 and www.thegame.com/video_id/movie=349587?speed=4. This example will *not* match www.thegame.com/playvideo/video_id=89779 because the video_id pattern does not match the beginning of the URI.

– The following is an example of how to add a host to the no cache list, using either a hostname or a host IP address:

```
<policy>
<no_cache_host_list>
    <host_no_cache>shop.offlineshoppppping.com</host_no_cache>
    <host_no_cache>202.202.1.16</host_no_cache>
</no_cache_host_list>
</policy>
```

The host_no_cache option supports wildcard configurations for both a hostname and a host IP address. The following example will match any host on the offlineshopping.com, any host that has a name with “offlineshopping.com” in the name, and any host that has an IP address that matches either 202.202.1.* or *.202.202.1.

```
<policy>
<no_cache_host_list>
    <host_no_cache>offlineshopping.com</host_no_cache>
    <host_no_cache>202.202.1</host_no_cache>
</no_cache_host_list>
</policy>
```

This example will match the following:

- aaa.offlineshopping.com
- 123456offlineshopping.com
- jjj.offlineshoping.com.au
- 122.202.202.1
- 202.202.1.3

– The following is an example of how to add a subnet to the no cache list. The <no_cache_subnet_list> section matches any IP address that is associated with either the client requesting the content or the server servicing the request. Therefore, if one of the IP addresses that is associated with a specific session falls into the subnet IP range specified in this policy, the information will not be cached-in or cached-out:

```
<policy>
<no_cache_subnet_list>
    <subnet_no_cache>192.168.0.150</subnet_no_cache>
    <subnet_no_cache>192.168.1.0/24</subnet_no_cache>
```

```
    <subnet_no_cache>192.168.2.150-
        192.168.2.158</subnet_no_cache>
</no_cache_subnet_list >
</policy>
```

- **Cache data expiration:** There are several system parameters that affect cache data expiration:
  - **expiration_high_water_mark:** This parameter determines how full the cache data storage can become, based on percentage, before the system performs an expiration task. By default, the system checks this value at 0400 system time to determine if an expiration task needs to be performed. If the cache disk volume usage is above this parameter, the expiration task begins to run. The default value for this parameter is 97%.
  - **expiration_low_water_mark:** This parameter controls at what point the system expiration task can stop deleting a bulk of objects from the cache data storage. Once the cache disk volume usage falls below this parameter, the expiration task stops running. The default value for this parameter is 85%.
  - **volume_critic_water_mark:** When the cache disk volume usage reaches the level that is defined by this parameter, as a percentage of disk space used, the application stops writing to this volume. The default value for this parameter is 98%.
  - **Busy window:** The busy window determines the time frame during which the expiration task does not run. At the end of the busy window, the system checks the expiration_high_water_mark parameter to see if an expiration task needs to be performed.
    - **–start_hour_for_busy_window:** This parameter determines the start time for the busy window. The default value for this parameter is 16 (1600 system time).
    - **–busy_window_size_in_hours:** This parameter determines the length of the busy window. You add this number of hours to the start_hour_for_busy_window to determine when the busy window has ended. The default value for this parameter is 12.

  Based on the default values of the start_hour_for_busy_window and the busy_window_size_in_hours parameters, at 0400 system time the system will determine whether the cache data storage is at or above the expiration_high_water_mark parameter. If the cache data storage is at or above the high water mark, then the expiration task determines the least recently hit (least popular) object in the storage, and removes it. After this removal, the task checks to see if the expiration_low_water_mark parameter has been reached. If the low water mark has *not* been reached, the expiration task continues to remove the least popular objects until the low water mark is reached.

  The cache data expiration process is performed for objects that have been cached-in. For objects that have gone through the VERIFY step, but are not yet cached-in (because they have been seen only once), the system will remove any record of hashes that have an age greater than that configured in the max_hours_hash_not_touched system parameter, for which the default is 24 hours. For more information on the max_hours_hash_not_touched system parameter, see the “Flush Hours” topic.

  To modify the cache data expiration parameters, add or edit the following text to the <service> subsection of the <common> section:

```
<expiration>
   <expiration_low_water_mark>low_water_mark_%</expiration_low_water_mark>
   <expiration_high_water_mark>high_water_mark_%</expiration_high_water_mark>
   <volume_critic_water_mark>volume_critical_water_mark_%</volume_critic_water_mark>
   <start_hour_for_busy_window>start_time_for_busy_window</start_hour_for_busy_window>
   <busy_window_size_in_hours>number_of_hours</busy_window_size_in_hours>
</expiration>
```

  For example:

```
<expiration>
    <expiration_low_water_mark>85</expiration_low_water_mark>
    <expiration_high_water_mark>92</expiration_high_water_mark>
    <volume_critic_water_mark>96</volume_critic_water_mark>
    <start_hour_for_busy_window>1400</start_hour_for_busy_window>
    <busy_window_size_in_hours>12</busy_window_size_in_hours>
</expiration>
```

Supporting Netflix

The Netflix protocol works differently on different devices. The supported devices in VDS TC are divided into two groups:

- iOS Apple devices, such as iPhones, IPads, and Apple TV
- Android OS devices, such as tablets, Smart Phones and Smart TVs, and Streamers

Configure the following to enable the VDS TC system to support the Netflix protocol:

- Add the following text to the configuration file in the <protocols> subsection of the <service> section:

  ```
  <enable-netflix>1</enable-netflix>
  <enable-silverlight>1</enable-silverlight>
  ```

- Add the following text to the <policy> subsection of the <service> section in the configuration file to configure support for the Netflix CDN IP ranges, where *CDN_IP_Range* is a CDN IP range that is appropriate for *your* installation.

  ```
  <netflix_cdn_subnet_list>
      <netflix_cdn_subnet>CDN_IP_Range</netflix_cdn_subnet>
  </netflix_cdn_subnet_list>
  ```

Note To add more than one CDN IP Range, add additional **<netflix_cdn_subnet>***CDN_IP_Range***</netflix_cdn_subnet>** lines.

For example:

```
<netflix_cdn_subnet_list>
<netflix_cdn_subnet>108.175.0.0/16</netflix_cdn_subnet>
<netflix_cdn_subnet>198.45.0.0/16</netflix_cdn_subnet>
</netflix_cdn_subnet_list>
```

Supporting Video Skips URL Strings Configuration

Cisco VDS TC handles video transactions as skips (jumps) if the URL includes some specific strings, such as: "start=, "begin=" that are found in the popular OTT video sites. Starting with Cisco VDS TC Release 5.2.0, you can add additional video skip/jump strings to match in addition to the default strings the application is seeking ("start=, "begin="). You should add only new string matches. You should not add the already existing default ones of "start=", "begin=".

You configure add additional video skip/jump strings within the <policy> subsection of the <service> section in the configuration file by adding the following configuration:

Note If there are additional parameters in the <policy> section, the <jump_string_match_list> should be the *first* element in this section.

```
<policy>
<jump_string_match_list>
<jump_string_match>relative_pos=#</jump_string_match>
</jump_string_match_list>
</policy>
```

This parameter and number represent a "jump" request in the URL, for example: seek_sec=61.

Note The <jump_string_match_list> structure can include up to 64 entries.

In addition, add the following value to the cluster_conf <policy> section:

```
<http_closed_file_timeout>25</http_closed_file_timeout>
```

System Load Monitoring

System load monitoring measures packet delays and packet loss, and if the packet delays or packet loss reach thresholds that you have configured, you can send an SNMP trap message or disable the service. You configure system load monitoring within the <policy> subsection of the <service> section in the configuration file.

- **check_overload_interval:** This field defines how frequently, in seconds, the NICs are polled for packet drops and packet delay.
- **overload_drop_percent:** This field defines a packet drop percentage threshold to monitor for, calculated per interface. If this threshold is exceeded a consecutive number of times, as defined in the <failed_overload_test> field, the overload action defined in the <overload_action> field occurs.
- **overload_packet_delay:** This field defines a packet delay threshold, in milliseconds, to monitor for, calculated across all interfaces. If this threshold is exceeded a consecutive number of times, as defined in the <failed_overload_test> field, the overload action defined in the <overload_action> field occurs.
- **failed_overload_test:** This field defines the number of consecutive times the packet drop threshold or the packet delay threshold must exceed their configured values to trigger the action defined in the <overload_action> field.
- **overload_action:** This field defines the action to take if the packet drop threshold or packet delay threshold has been exceeded for the configured number of consecutive times. This value can be NOTHING, TRAP_ONLY, or DISABLE.

The following example will poll interfaces every 12 seconds. If in three consecutive polls the delay was more than 500 ms or if a packet loss is more than 1.22%, the service is disabled:

```
<policy>
    <check_overload_interval>12</check_overload_interval>
    <overload_drop_percent>1.22</overload_drop_percent>
    <overload_packet_delay>500</overload_packet_delay>
    <failed_overload_test>3</failed_overload_test>
    <overload_action>DISABLE</overload_action>
</policy>
```

Platform Specific Features

- **Cluster File System (CFS):** CFS is a distributed file-system that can operate seamlessly over *n* times storage devices. This provides a very large storage for each cache engine, enabling very fast data retrieval of cached data. The CFS is a content aware file-system, optimized specifically for the content it stores. It uses less I/O operations to service the amount of cached information it serves. Faster data throughput is achieved. When VDS TC operates in an Integrated Appliance or Cluster-ready VDS TC configuration, the CFS is used only by the active cache-engine (in cases where the platform is equipped with only one cache-engine).

Platform Operational Specific Features

- **Configuring an NTP server and time zone:** Follow these steps to configure the VDS TC management server to use an NTP server and configure the time zone for the VDS TC system:

  Add the following text to the configuration file in the <common> section:

```
<ntp>
    <server-ip>ntp-server1-ip</server-ip>
    <server-ip>ntp-server2-ip</server-ip>
    <timezone>GMT <+/->offset</timezone>
</ntp>
```

  where

  – *ntp-server1-ip* is the IP address or hostname of the first NTP server to use

Note To use the local server as the NTP server, enter **127.127.1.0** for the IP address.

  – *ntp-server2-ip* is the IP address or hostname of a second NTP server to use, if desired

  – *offset* is the GMT offset (+/-), for the time zone of the VDS TC system.

  For example:

```
<ntp>
    <server-ip>1.asia.pool.ntp.org</server-ip>
    <server-ip>2.asia.pool.ntp.org</server-ip>
    <timezone>GMT+3</timezone>
</ntp>
```

Note The GMT *offset* value only supports a whole number between 0 and 12. All other time zones are NOT supported.

Note The time will not automatically adjust for daylight savings time (DST). If you need to adjust the time for DST, you will need to make this change manually.

Note If you upgrade to VDS TC 5.7.3, you may need to reconfigure the NTP settings.

Note When you use the **date** command from the VDS TC manager operating system, you may see that the GMT offset appears opposite of what you entered. For example if you entered **<timezone>GMT+8</timezone>** in the configuration file, when you enter the **date** command in the operating system, you will see "Tue August 1 14:16:35 GMT-8 2014". The output in the date command is intentionally reversed for backwards compatibility with POSIX standards.

- **Duplicate logs to external syslog:** This feature duplicates the local syslog information of the VDS TC system and sends it to an external syslog server while the system is running.

  To enable this feature add the following text to the configuration file, where the *IP_address* parameter is the IP address of the external syslog server:

  ```
  <mgmt-config>
      <external_syslog_ip>IP_address</external_syslog_ip>
  </mgmt-config>
  ```

  You can also stop and start forwarding the syslog to an external syslog server using the CLI. *See* eventlog, page 4-24.

- **Enabling Hardware Traps:** Enabling hardware traps enables you to see hardware traps in Cisco VDS Transparent Caching Manager. The equipment traps must also be configured in the target device. By default this feature is disabled.

  To enable hardware traps, add the following text to the <mgmt-config> section of the configuration file:

  ```
  <enable-equipment-traps>1</enable-equipment-traps>
  ```

- **Management Access Control List (ACL):** This feature enables you to configure a white list or a black list to determine what source IP addresses are allowed to access the VDS TC management interfaces (the CLI and the web management interface).

  – **White list:** If you use a white list, enter the IP addresses that should be permitted to access the VDS TC management interfaces. Any IP addresses that do not match an entry in the white list are denied access to the management interfaces.

  – **Black list:** If you use a black list, enter the IP addresses that you do not want to have access to the VDS TC management interfaces. Any IP addresses that do not match an entry in the black list are permitted access to the management interfaces.

  To use a white list to determine who should be allowed access to the VDS TC management interfaces, add the following text to the <mgmt-config> section of the configuration file, where the *permitted_IP_address* parameter is the IP address of a host that should be permitted access to the VDS TC management interfaces:

  ```
  <access_lists>
      <white_access_list>
          <access_entry>permitted_IP_address</access_entry>
          <access_entry>permitted_IP_address</access_entry>
      </white_access_list>
  </access_lists>
  ```

  For example:

  ```
  <access_lists>
      <white_access_list>
          <access_entry>192.168.1.1</access_entry>
          <access_entry>192.168.1.2</access_entry>
      </white_access_list>
  ```

```
</access_lists>
```

Note Any IP addresses that do not match an entry in the white list are implicitly denied.

To use a black list to determine who should be allowed access to the VDS TC management interfaces, add the following text to the <mgmt-config> section of the configuration file, where the *denied_IP_address* parameter is the IP address of a host that should be denied access to the VDS TC management interface:

```
<access_lists>
    <black_access_list>
        <access_entry>denied_IP_address</access_entry>
        <access_entry>denied_IP_address</access_entry>
    </black_access_list>
</access_lists>
```

For example:

```
<access_lists>
    <black_access_list>
        <access_entry>80.122.12.1</access_entry>
        <access_entry>80.122.12.2</access_entry>
    </black_access_list>
</access_lists>
```

Note Any IP addresses that do not match an entry in the black list are implicitly permitted.

- **Forwarding SNMP traps for the VDS TC application:** The VDS TC caching service generates SNMP traps for certain events. You can configure the system to forward these traps to an external server.

  Follow these steps to control the forwarding of VDS TC traps:

Step 1 If you do not already have one, obtain an SNMP monitoring system.

Step 2 Add the following text to the configuration file in the <common> section:

```
<snmp>
<trap-ip>IP_Address_SNMP_Server</trap-ip>
<snmp-read-community>read_community_string</snmp-read-community>
<snmp-write-community>write_community_string</snmp-write-community>
<snmp-trap-community>trap_community_string</snmp-trap-community>
</snmp>
```

For example:

```
<snmp>
    <trap-ip>10.11.12.1</trap-ip>
    <snmp-read-community>gdcbhv</snmp-read-community>
    <snmp-write-community>nkppui</snmp-write-community>
    <snmp-trap-community>ffff</snmp-trap-community>
</snmp>
```

- **Email alerts:** In addition to SNMP traps, you can forward critical VDS TC alerts to a specific email server and email address.

To configure the forwarding of critical alerts to an email address, add the following text to the configuration file where the dns_server parameter points to a DNS server that can resolve the domain name to which the email should be sent:

```
<mgmt-config>
    <nameserver>dns_server</nameserver>
    <alert-email>email_address</alert-email>
</mgmt-config>
```

For example:

```
<mgmt-config>
    <nameserver>194.90.1.5</nameserver>
    <alert-email>support@cisco.com</alert-email>
</mgmt-config>
```

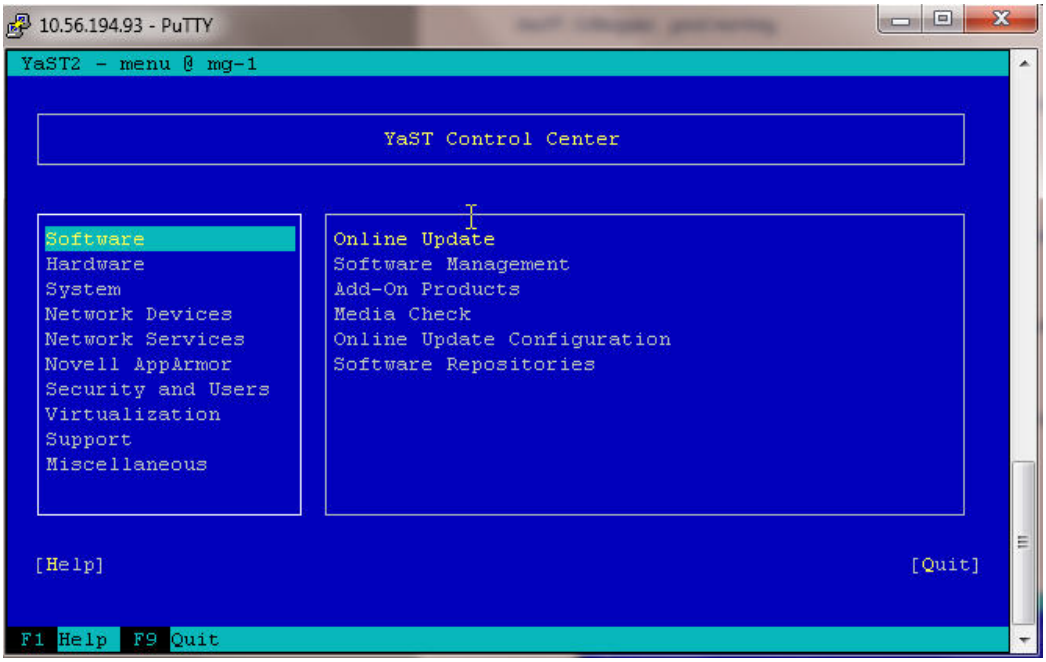
Note If the VDS TC Manager needs to use an SMTP e-mail relay, you must also perform the following steps:

Step 1 Using SSH software, log into the VDS TC management server using the username **padmin** and the password that was provided by Cisco.

Step 2 Enter the **su root** command to switch to the root user. Enter the password that was provided by Cisco.

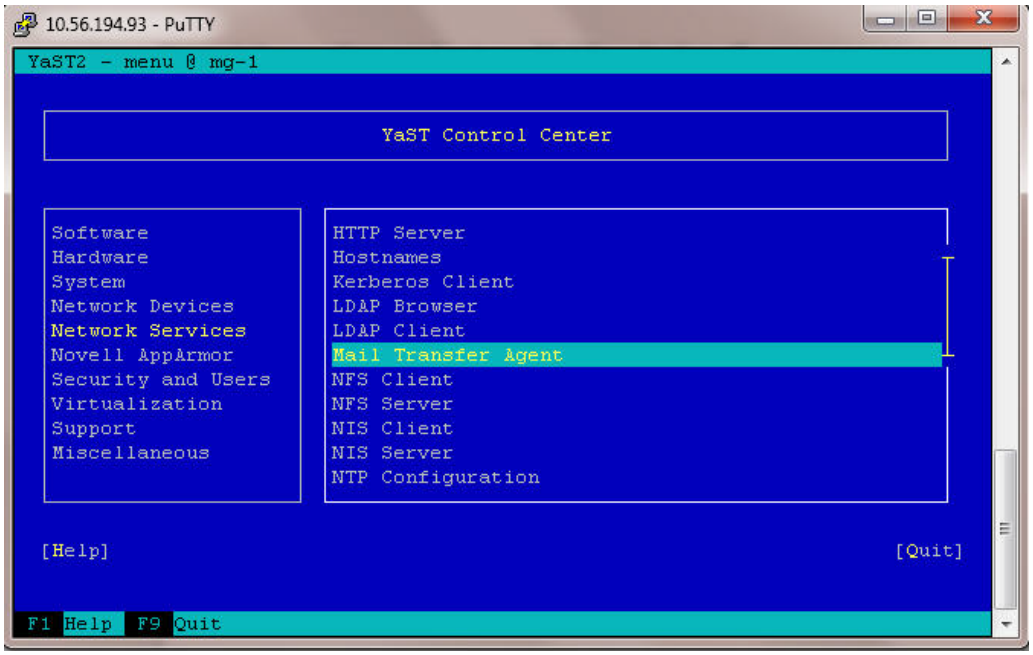
Step 3 Enter the command **yast** to enter the YaST environment.

Figure 3-1YaST Control Center



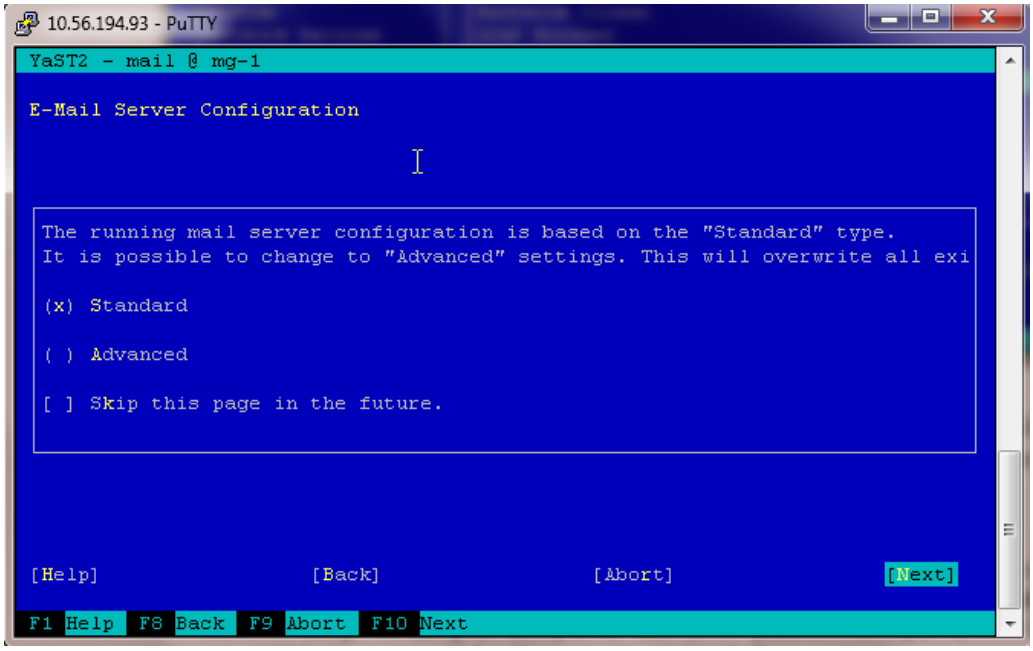
Step 4 From the YaST Control Center window choose **Network Services** from the left pane and then choose **Mail Transfer Agent** from the right pane. Press **Enter**.

Figure 3-2 Network Services - Mail Transfer Agent



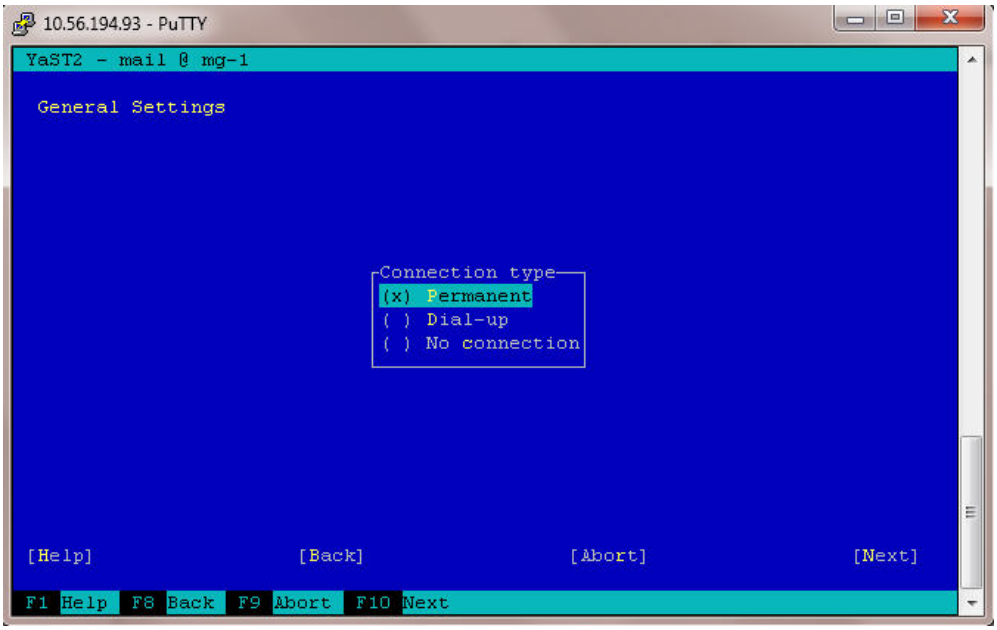
Step 5 The E-Mail Server Configuration window appears. From this window if Standard is not selected (an X will appear in front of the option if it is selected), press **Alt-S** to select it and then press **Alt-N** to proceed to the next window.

Figure 3-3 E-mail Server Configuration



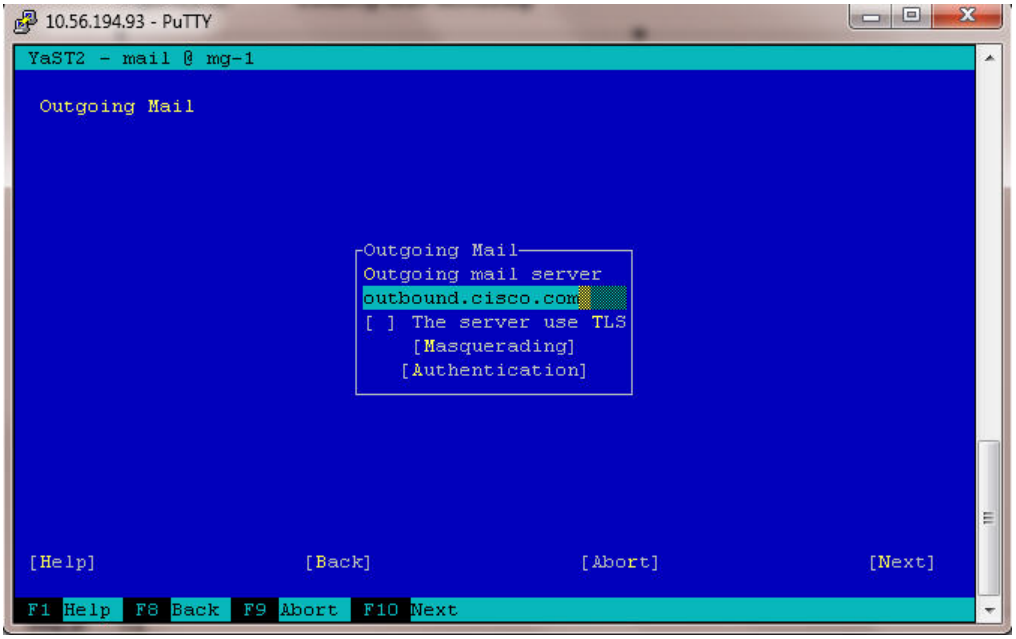
Step 6 The General Settings window appears. From the Connection Type section, if Permanent is not selected, press **Alt-P** to select it and then press **Alt-N** to go to the next window.

Figure 3-4 General Settings Window



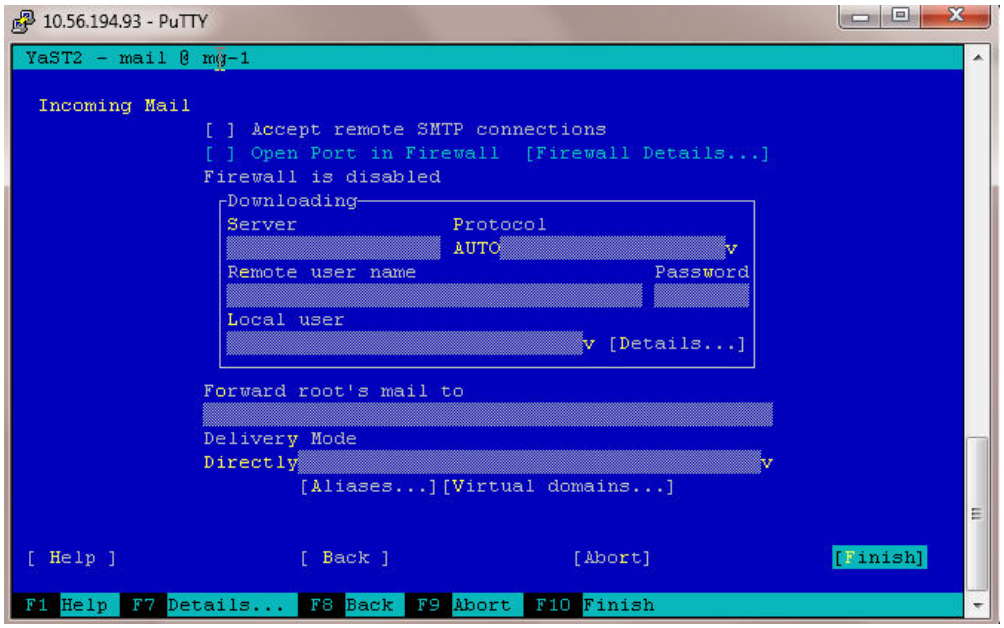
Step 7 The Outgoing Mail window appears. From this window press **Alt-O** to select the Outgoing Mail Server text box and enter the FQDN of the server to use as the email relay server, for example outbound.cisco.com. Press **Alt-N** to move to the next window.

Figure 3-5 Outgoing Mail Window



Step 8 The Incoming Mail window appears. Do not make any changes in this window. Press **Alt-F** or **F10** to finish and save the configuration.

Figure 3-6 Incoming Mail Window



Step 9 From the YaST Control Center window, press **F9** or **Alt-Q** to quit the program.

Step 10 Exit from the SSH window.

- **Export CDR:** The platform tracks each cache-out session and writes out a CDR record that is related to the session when it ends. The CDR recording files are created periodically and can be retrieved using FTP using an 'anonymous' user-id. No password is required, even if prompted. The set of CDR files are created and managed periodically.

  The following is an example of a CDR record:

```
14-05-15
15:58:32.000,HTTP,9D8542B1D54BEC17435272B7F6507A76F06B8B2D,10.201.19.11,/?Transaction-Prof
ile=CO_5MB,SESSION_ENDED,9997236,6,10.109.55.7,54090,10.201.19.11,80,0,0,CACHE_OUT,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,,,,,,,,,,,51DF9719,10000000,10000000,6,CDR_EOL
```

Traffic Specific Features

Starting with VDS TC Release 5.2.0, the ToS/DSCP configuration is configured using the policy manager in VDS TC Manager. Starting with VDS TC Release 5.1 the <tos_markup> is not valid in the cluster configuration file.

Fine Tuning System Behavior for Mobile Operators

The VDS TC platform has a built-in mechanism that is able to adjust to the network behavior changes because of network delays and network congestions. You must configure all of the following parameters in the <tcp> subsection of the <service> section in the configuration file to not lose packets on transmission. An explanation for each parameter is provided in Table 3-1.

```
<tcp>
    <conn_idle_timeout>60</conn_idle_timeout>
    <tcp_snd_buf>393216</tcp_snd_buf>
    <min_cwnd_in_packets>3</min_cwnd_in_packets>
    <min_ssthresh_in_packets>15</min_ssthresh_in_packets>
    <slow_start_reduce_ratio>3</slow_start_reduce_ratio>
    <http_timeout_sec>60</http_timeout_sec>
</tcp>
```

Table 3-1 TCP parameters

Parameter	Definition
<conn_idle_timeout>*seconds***</conn_idle_timeout>**	The seconds after which the connection is removed from the internal connection table, as a result of no activity on that session. In the example this is 60 seconds.
<tcp_snd_buf>*no_bytes***</tcp_snd_buf>**	The maximum number of bytes that the caching application sends (in cache-out) before receiving any acknowledgment from the client (in flight bytes). In the example this is 393,216.
<min_cwnd_in_packets>*no_pkts***</min_cwnd_in_packets>**	The minimum number of packets to which the TCP congested window can drop. This parameter affects only the cache-out sessions. The higher the number, the more packets that can be sent even if the client did not receive them or is congested. In the example this is 3.
<min_ssthresh_in_packets>*no_pkts***</min_ssthresh_in_packets>**	The minimum number of packets to which the TCP slow start boundary can drop. In the example this is 15. This parameter affects only cache-out sessions. The higher the number, the faster a TCP connection can recover after packet loss. The slow_start_reduce_ratio parameter determine the factor that the caching application should reduce.

Table 3-1 TCP parameters (continued)

Parameter	Definition
<slow_start_reduce_ratio>*factor***</slow_start_reduce_ratio>**	This configures the factor that the caching application can use to reduce the TCP slow start threshold upon packet loss (in cache-out mode only). The number is a factor of ten percent. For example, 5 would set the ratio to 50%, which is the default TCP behavior. The maximum value is ten. The higher the number, the faster a TCP connection recovers from a packet loss. In the example this is 3, which would set the factor to 30%.
<http_timeout_sec>*seconds***</http_timeout_sec>**	This value defines after how many seconds of no activity the session is removed from the internal session table. The default is 20 seconds.
<use_new_stack_params>1**</use_new_stack_params>**	This parameter is not currently supported.
<scaling_window_multiplier>*multiplier***</scaling_window_multiplier>**	This parameter is not currently supported.

Controlling Core Dumps

A caching application core dump is when the caching application writes everything from its memory to disk. This can consume a lot of disk space, especially as the memory for the caching application is increased.

The default behavior is for the caching application to perform a core dump. To configure the caching application to not perform a core dump, add the following text to the configuration file in the <policy> subsection of the <service> section:

```
<execute_debug_info>0</execute_debug_info>
```

Setting the value to 0 disables the core dump. To re-enable the core dump feature, set the value to **1**.

Configurations Using the CLI

The following sections describe commands that you can use from the VDS TC CLI to configure the VDS TC device.

This section includes the following information:

- Configuring Passwords
- Recovering Passwords
- Configuring the Management Network
- Configuring Time

- Managing the Caching Service
- Resetting the Management Service
- Managing Traffic Detection

Caution The commands and their options are case sensitive.

Configuring Passwords

To set or change the passwords to access regular mode and enable mode, use the following commands:

Command	Purpose
console# **access user-password** *new_password*	Establishes a new password or changes an existing password for the regular command level, also referred to as the user password.
console# **access enable-password** *new_password*	Establishes a new password or changes an existing password for the enable mode.

Recovering Passwords

If you forget the initial CLI password or the enable mode password, you can reset them to their default values using the special Rescue CLI. The Rescue CLI is available only from the serial console. For more information, *see* CLI Modes, page 2-3.

Configuring the Management Network

To be able to access the CLI using SSH and a LAN connection, you must configure the IP address and default gateway address for the management server. You can configure an IPv4 address, an IPv6 address, or both. In enable mode, follow these steps to configure this information:

Table 3-2 Configuring a Management IPv4 Address:

	Command	Purpose
Step 1	console# **config**	Enters Configuration mode.
Step 2	configuration# **network ip** *IP_address netmask*	Configures the management IPv4 address of the VDS TC device. You must enter the subnet mask using the dotted decimal notation.
Step 3	configuration# **network default_gw** gateway_address	Configures the default IPv4 gateway address for the VDS TC to use.
Step 4	configuration# **apply**	Applies the configurative changes.
Step 5	configuration# **exit**	Returns to enable mode.

	Command	Purpose
Step 6	console# **ping** [**-c** *count*] [-I Source_*interface_or_sourceIP_address*] *destination_IP*	Tests network connectivity. After you configure the network settings, you should test the configuration. This command is available in both regular and enable mode. **Note** If you do not specify the number of times to repeat the ping with the count option, the ping will continue until you press **Ctrl-C**.
Step 7	console# **traceroute** *destination_IP*	Test network connectivity and displays the router hops that packets will actually take when traveling to their destination. This command can help troubleshoot network connectivity and is a good troubleshooting step if the ping fails.

Table 3-3 Configuring a Management IPv6 Address

	Command	Purpose
Step 1	console# **config**	Enters Configuration mode.
Step 2	configuration# **network ip6** *ipv6_address/ipv6_prefix*	Configures the management IPv6 address of the VDS TC device.
Step 3	configuration# **network default6_gw** *gateway_address*	Configures the default IPv6 gateway address for the VDS TC device to use.
Step 4	configuration# **apply**	Applies the configurative changes.
Step 5	configuration# **exit**	Returns to enable mode.
Step 6	console# **ping** [**-c** *count*] [-I Source_*interface_or_sourceIP_address*] *destination_IP*	Tests network connectivity. After you configure the network settings, you should test the configuration. This command is available in both regular and enable mode. **Note** If you do not specify the number of times to repeat the ping with the count option, the ping will continue until you press **Ctrl-C**.
Step 7	console# **traceroute** *destination_IP*	Test network connectivity and displays the router hops that packets will actually take when traveling to their destination. This command can help troubleshoot network connectivity and is a good troubleshooting step if the ping fails.

Configuring Time

NTP is a networking protocol that is used to synchronize clocks. You can use NTP to configure computer systems with the IP address of an NTP time source server.

You can manually configure the local time on VDS TC management server or you can configure the VDS TC management server to point to an NTP server.

Manually Configuring Time

To manually configure the time on the VDS TC management server, use the following command in Configuration mode:

Command	Purpose
`configuration# time MMDDYYhhmm`	Sets the system date and time. Two numbers are used to represent each part: month, day, year, hour, and minutes. For example **time 1104120815** would set the date to November 4, 2012 8:15 am.

Note The CLI configuration# prompt requires Enable mode privileges. For more information, *see* Chapter 4, "Using CLI Commands".

To verify that the time that is currently set on the VDS TC device, enter **show time** in either regular mode or enable mode.

Using an NTP Server

To configure the VDS TC management server to use an NTP server, refer to the "Configuring an NTP server and time zone" in the Platform Operational Specific Features section.

Managing the Caching Service

From the CLI you can stop or start the caching service. To manage the caching service use the **oper service** command in enable mode.

oper service {**powerdown** | **powerup** | **stop** | **start**}

Syntax Description

powerdown	Service power down.
powerup	Service power up.
stop	Stops the caching service.
start	Starts the caching service after it has been stopped.

Resetting the Management Service

In Configuration mode, only one user at a time can perform a configuration operation.

If your active terminal session does not respond because another user is already performing a configuration operation, you can reset the management service to recover the ability to configure the system.

Caution Use this option with caution. If you reset the management service, all open CLI sessions are reset, including the session from which you are executing the command, and any configurations that were not applied will be lost.

To reset the management service use the following command in enable mode:

Command	Purpose
`console# reset`	Resets all active CLI sessions in the system.

Managing Traffic Detection

From the CLI you can manage traffic detection by categorizing traffic types. *See* Working with the Configuration Files, page 2-5.

File-Based Configuration

When configuring the software settings in the main configuration file, you can configure fields in the <common> section or in the <blade id=1> section. When there is a value for a field in both sections, the value of the field in the <blade id=1> section takes priority over the value of the field in the <common> section.

Note Any changes that you make to the VDS TC configuration file will only take affect after you import the configuration file into the VDS TC management server. For information on how to import this file, see Working with the Configuration Files in Chapter 2, "Working with Cisco VDS TC Management Tools".

This section includes the following configuration information:

- Configuring SNMP
- Configuring P2P Protocols
- Configuring Bandwidth Management
- Configuring Caching Policies
- Configuring Virtual IP Address
- NIC Flapping Option
- Applying Configuration Changes
- Upgrading the VDS TC Software
- Updating the VDS TC License

Configuring SNMP

To configure the SNMP settings add or edit the following fields in the <common> section of the configuration file:

```
<snmp>
    <trap-ip>IP_Address_SNMP_Server</trap-ip>
    <snmp-read-community>read_community_string</snmp-read-community>
    <snmp-write-community>write_community_string</snmp-write-community>
    <snmp-trap-community>trap_community_string</snmp-trap-community>
</snmp>
```

For example:

```
<snmp>
<trap-ip>10.11.12.1</trap-ip>
<snmp-read-community>gdcbhv</snmp-read-community>
<snmp-write-community>nkppui</snmp-write-community>
<snmp-trap-community>ffff</snmp-trap-community>
</snmp>
```

Configuring P2P Protocols

To enable or disable the different P2P protocol support, add or edit the following fields in the <common> section or <blade id=1> section, in the <service> <protocols> subsection of the configuration file:

- BitTorrent

  <enable-bittorent>*value***</enable-bittorent>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.
- BitTorrent uTP

  <enable-utp-bittorent>*value***</enable-utp-bittorent>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.

Note Currently uTorrent 3.4.2 is supported.

- eDonkey

  <enable-edk>*value***</enable-edk>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.
- Ares

  <enable-ares>*value***</enable-ares>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.
- HTTP

  <enable-http>*value***</enable-http>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.

Note

For example, to enable Ares support, add the following to the <common> section or <blade id=1> section of the configuration file:

```
<enable-ares>1</enable-ares
```

Configuring Bandwidth Management

To configure bandwidth management, add or edit the following fields in the <common> section or <blade id=1> section, <service> subsection of the configuration file:

```
<bandwidth-management>
    <enable-bandwidth-management>value<enable-bandwidth-management>
    <bandwidth-per-connection>max_bw_per_IP<bandwidth-per-connection>
</bandwidth-management>
```

Replace *value* with **1** to enable the bandwidth management service or **0** to disable the bandwidth management service. Replace *max_bw_per_IP* with the maximum bandwidth per connection, in bytes/second, that you want to support.

Traffic Forwarding Modes

The traffic forwarding mode controls whether the VDS TC system will forward traffic in promiscuous mode or bounce mode:

- **Promiscuous:** The L2/L3 switch forwards traffic via two dedicated ports without changing L2 addresses (as-is).
- **Bounce:** The platform sends packets back using the same interface while swapping the source and destination MAC addresses.

To configure this setting, edit the following field in the <common> or <blades>1 section, <service> <net> subsection of the configuration file:

```
<fwd-mode>mode</fwd-mode>
```

Replace *mode* with either **BOUNCING** or **PROMISC**.

Configuring Caching Policies

The caching policy indicates the percentage of upstream P2P traffic that must come from the internal cache. To configure this setting, add or edit the following fields in the <service> section of the configuration file:

```
<policy>
    <upload_cache_out>%_of_traffic</upload_cache_out>
</policy>
```

Replace %_of_traffic with one of the following:

- **0:** Disables this feature. All upstream traffic can come from local peers.
- **1-99:** This is the specified percentage of the upstream traffic that must come from the VDS TC cache storage and the remainder of the upstream traffic can come from local peers.
- **100:** Upstream traffic can only come from the internal cache.

Configuring Virtual IP Address

The VDS TC caching engine has multiple virtual IP addresses that are used by the L3 switch for health monitoring, load sharing, and next hop addresses. The L3 switches query the virtual IP addresses with ICMP requests to verify health of each interface and the overall server availability. You can configure one virtual IP address per interface, which enables you to load distribute redirected traffic between the different interfaces of the caching engine.

The virtual IP addresses that you configure on the VDS TC host must match the IP addresses on the MLS DPI switch. To configure the virtual IP address of an interface, add or edit the following field under the <blade id=1> section, <cache-engine><network><network_interfaces> <nic nic_index=> subsection of the interface:

```
<vip>IP_address</vip>
```

For example:

```
<nic nic_index="0">
    <name>eth4</name>
    <nic_detail>IFF_PF_PACKET</nic_detail>
    <vip>10.138.201.1</vip>
</nic>
```

NIC Flapping Option

You can configure the VDS TC system to initiate flapping whenever the system starts. To configure the NIC flapping option, add or edit the following in the <service> section of the configuration file:

```
<interface_flapping>
    <set_sleep_interfaces>1</set_sleep_interfaces>
    <shut_down_iff_delay>delay_1</shut_down_iff_delay>
    <start_up_iff_delay>delay_2</start_up_iff_delay>
</interface_flapping>
```

Enter a value in seconds for *delay_1* and *delay_2* to configure a shut_down_iff delay value and a start_up_iff delay value, respectively. The default value for the shut_down_iff delay is 10 seconds and the default value for the start_up_iff delay is 5 seconds.

The the shut_down_iff delay and start_up_iff delay values are used as follows when you configure the NIC flapping option:

Step 1 When the system starts, the system waits for a period of time equal to the shut_down_iff delay value and then it performs an ifdown for all interfaces that are currently up.

Step 2 The system then waits for a period of time equal to the start_up_iff delay value and then it does an ifup to all interfaces that were up in Step 1.

The following example activates flapping with the conditions of starting and restarting:

```
<interface_flapping>
    <set_sleep_interfaces>1</set_sleep_interfaces>
    <shut_down_iff_delay>3</shut_down_iff_delay>
    <start_up_iff_delay>5</start_up_iff_delay>
</interface_flapping>
```

Applying Configuration Changes

After configuring the configuration file as required, you can activate the new configuration.

During the activation process, VDS TC verifies the new configuration, and flags any errors found in the log.

Follow these steps to activate the new configuration file:

Step 1 Use VDS TC Manager to upload the new configuration file to VDS TC. *See* the *Cisco Videoscape Distribution Suite Transparent Caching Manager User Guide* (part number OL-28017-02) for more details.

Step 2 Open an SSH session to the VDS TC CLI and log in using the username **admin** and password.

Step 3 At the CLI prompt, enter **enable** to enter Enable mode.

Step 4 At the CLI prompt, enter **config** to access Configuration mode.

Step 5 At the CLI prompt, enter **import localhost** *filename* where *filename* is the name of the new configuration file that you uploaded in Step 1.

Note After performing this step, the new configuration is downloaded to the system, but it is not yet applied. The new configuration is applied only after you complete all of the activation steps and the services are restarted.

Step 6 The following options are available for displaying, applying, or discarding the new configuration, or for restoring an old configuration:

Command	Purpose
`console# show config`	Displays the currently loaded configuration.
`configuration# display`	Displays the new configuration that was loaded, but not yet applied.
`configuration# diff`	Displays the differences between the current configuration and the new configuration.
`configuration# apply`	Applies the new configuration in place of the current configuration.
`configuration# discard`	Discards the new configuration without making any changes to the current configuration.
`configuration# restore`	Restores the old configuration after applying a new configuration.

Step 7 After the **apply** command displays that the HTTP service has finished restarting, exit the configuration mode using the **exit** command.

Note The new configuration is applied only after all the activation steps are completed, and the service is restarted.

Step 8 Execute the following commands in Enable mode to stop and restart the VDS TS caching service:

a. **oper service stop**

b. **oper service start**

Step 9 Log off of the VDS TC CLI by entering the **exit** command twice.

The configuration session is complete.

Also *see* Working with the Configuration Files, page 2-5.

Upgrading the VDS TC Software

To upgrade the VDS TC software to VDS TC Release 5.7.3, please see the Cisco *Videoscape Distribution Suite Transparent Caching Application Upgrade Guide*, available at http://www.cisco.com/c/dam/en/us/td/docs/video/videoscape/distribution_suite/vds/v5_7_3/VDS-TC_5.7.3_app_upgrade_guide.pdf

Updating the VDS TC License

To view information about your license from the CLI, in Enable mode enter **show license**. Information about the installed license is displayed, including the version number and enabled features.

Follow these steps to install a new software license:

Caution If you are also upgrading the VDS TC software, follow the steps in the Cisco Videoscape Distribution Suite Transparent Caching Application Upgrade Guide for updating the VDS TC License. These steps are to update a license to add additional features or limits.

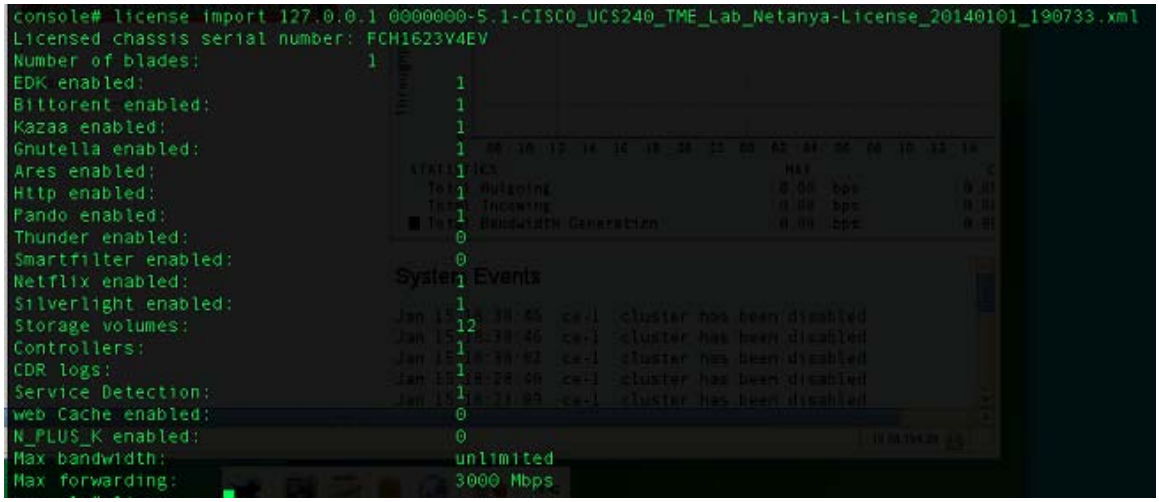
Step 1 Copy the new license file to your TFTP server.

Step 2 From the VDS TC prompt, enter the **enable** command. When prompted, enter the Enable mode password and press **Enter**.

Step 3 At the CLI in Enable mode, enter **license import** *TFTP_server filename*, where *TFTP_server* is the IP address of your TFTP server and *filename* is the name of the new license file, for example 0000000-5.7-CISCO_UCS240_XY-License_20140127_19862.xml.

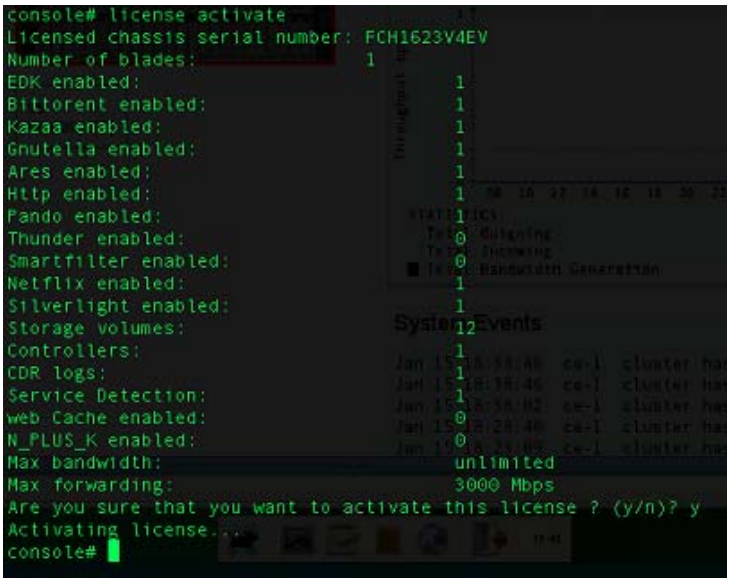
Note You must be in Enable mode, *not* Configuration mode to import a new license.

Figure 3-7 License Import



Step 4 Enter **license activate** to apply the license.

Figure 3-8 License Activation



Step 5 Use VDS TC Manager to confirm the upgrade and version number. In a web browser, enter the management IP address of the VDS TC appliance to connect to the VDS TC Manager.

Step 6 Enter a username of **padmin** and the password that was provided by Cisco.

Step 7 Choose **Configuration > License Manager** to confirm the license has been upgraded.

Configuring TACACS+ on the Server

TACACS+ is a an access control network protocol that allows user authentication and authorization with the customer's TACACS+ server for both the VDS TC Manager environment and the VDS TC CLI (using Telnet or SSH).

To be able to work with TACACS+ authentication and authorization, you must configure several parameters on the TACACS+ server and in the VDS TC management environment.

Note If the configured TACACS+ server is not available, or if a TACACS+ server is not configured, authentication and authorization is performed using the standard VDS TC management server username and password built-in mechanism.

Configuring TACACS+ for VDS TC Support, Using Cisco Secure ACS Release 4.x

To configure TACAC+ support for the VDS TC platform on the Cisco Secure ACS server, perform the following steps on the Cisco Secure ACS server:

Step 1 Create a network device group for the VDS TC systems:

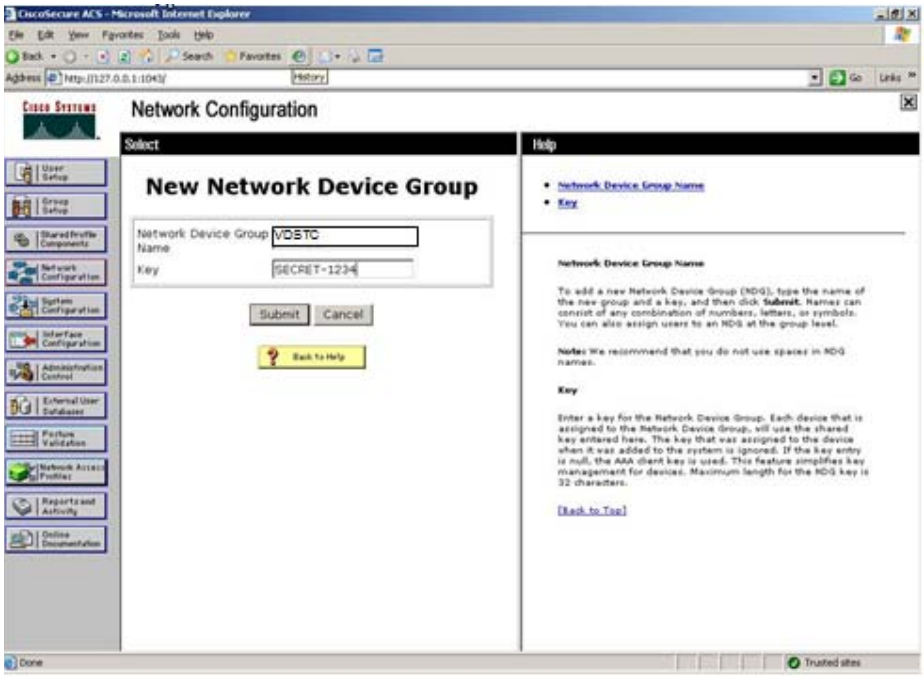
a. Click **Network Configuration**.

b. Click **Add Entry** to add a new network device group.

c. Enter a name for the network device group and in the Key field enter the shared key.

d. Click **Submit**.

Figure 3-9 Cisco Secure ACS New Network Device Group



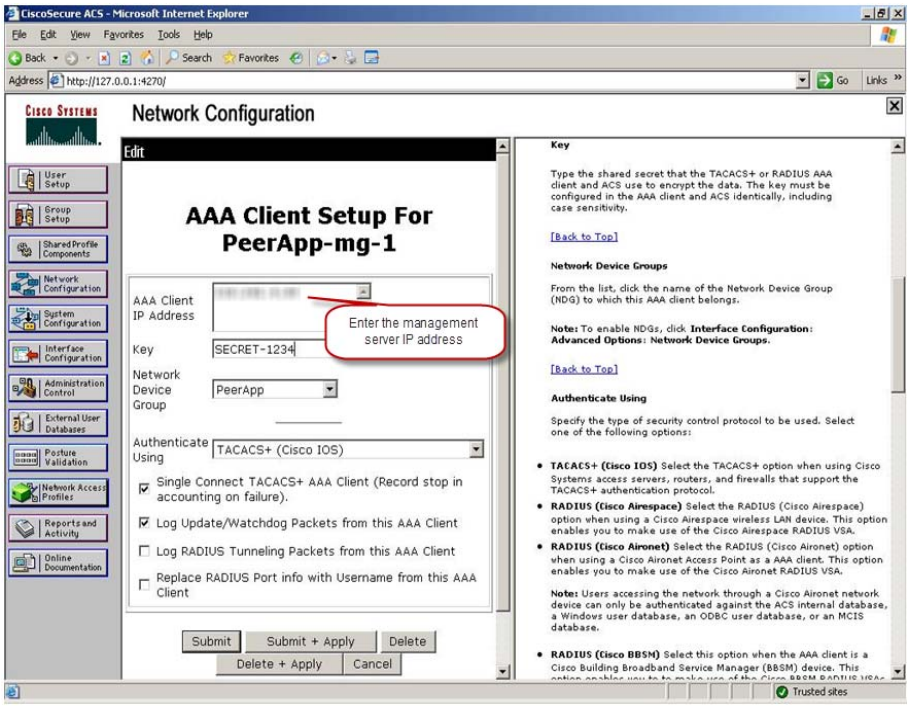
Step 2 For each VDS TC system that will use TACAC+, create a AAA client, including the management IP address of the VDS TC system and a shared secret (password):

a. Click **Network Configuration**.

b. Click the network device group you created in Step 1 and click **Add Entry**.

c. In the Add AAA Sever window, enter the IP address of the VDS TC system, enter the shared secret key, and choose **TACACS+(Cisco IOS)** from the Authenticate Using drop-down list.

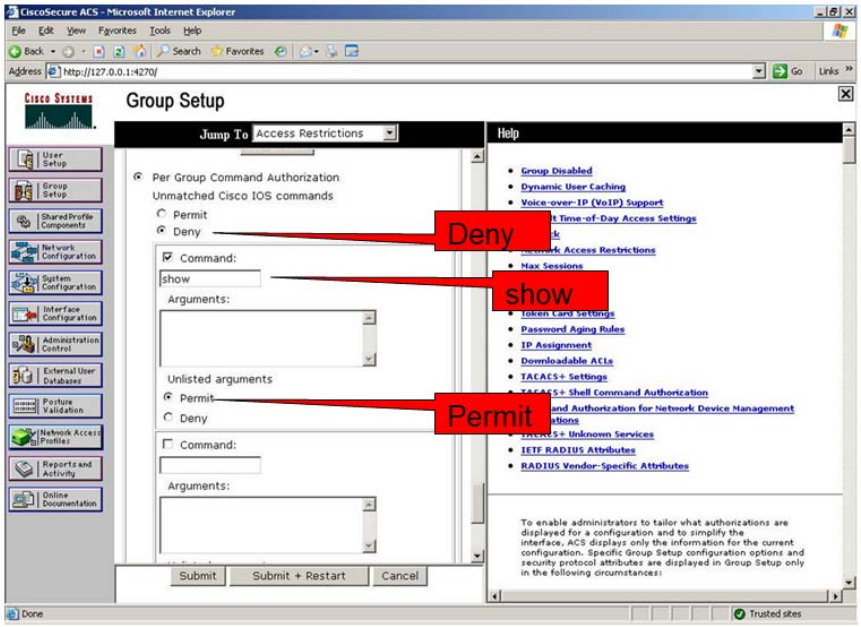
Figure 3-10 Cisco Secure ACS AAA Client Setup



Step 3 Configure authorization for the users. Each user in the TACACS+ server that should have access to VDS TC must be assigned to one of the following groups. You must create these groups on the TACACS+ server, with the following parameters.

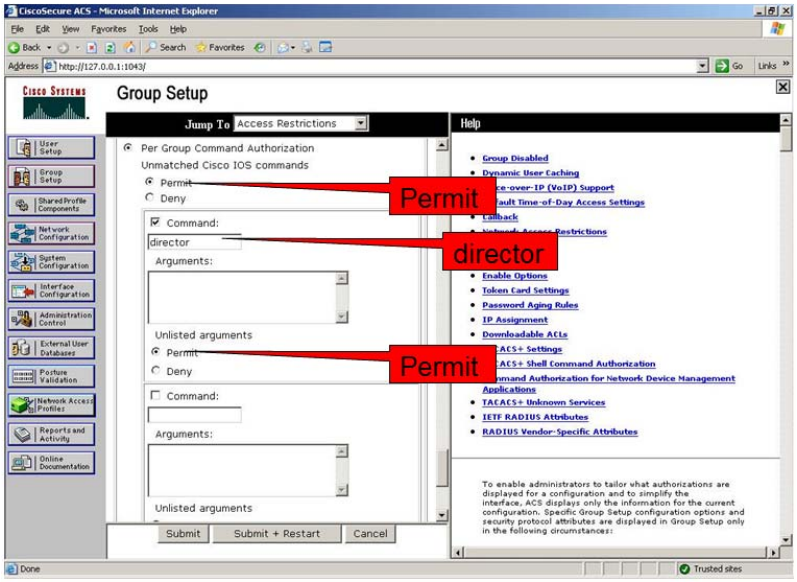
- **VDSTC-standard:** A user that is associated with this group can be used for standard VDS TC CLI login and standard VDS TC Manager login. For the CLI, it allows the user access to non privileged commands. It also allows the user access to the standard VDS TC Manager user interface. Configure the following settings for this group.
  - Group name: **VDSTC-standard**
  - Set the Per Group Command Authorization for this group to **Deny**.
  - Check the **Command** check box for this group and enter **show** in the command text box.
  - For the Unlisted Arguments setting, click **Permit**.

Figure 3-11 Cisco Secure ACS Group Setup: VDSTC-standard



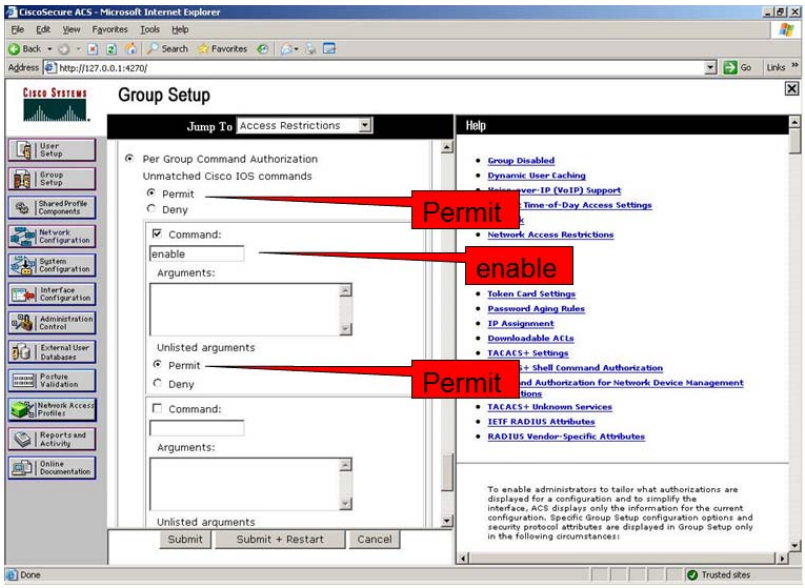
- **VDSTC-director:** A user that is associated with this group can modify the content of the caching application policies, managed inside the VDS TC Manager. Users that are not a member of this group will have view-only privileges for the caching application policy environment and will not be able to modify the policies. Configure the following settings for this group.
  - Group name: **VDSTC-director**
  - Set the Per Group Command Authorization for this group to **Permit**.
  - Check the **Command** check box for this group and enter **director** in the command text box.
  - For the Unlisted Arguments setting, click **Permit**.

Figure 3-12 Cisco Secure ACS Group Setup: VDSTC-director



- **VDSTC-privileged:** A user that is associated with this group can be used for Enable mode CLI command access and VDS TC Manager login. Configure the following settings for this group.
  - Group name: **VDSTC-privileged**
  - Set the Per Group Command Authorization for this group to **Permit**.
  - Check the **Command** check box for this group and enter **enable** in the command text box.
  - For the Unlisted Arguments setting, click **Permit**.

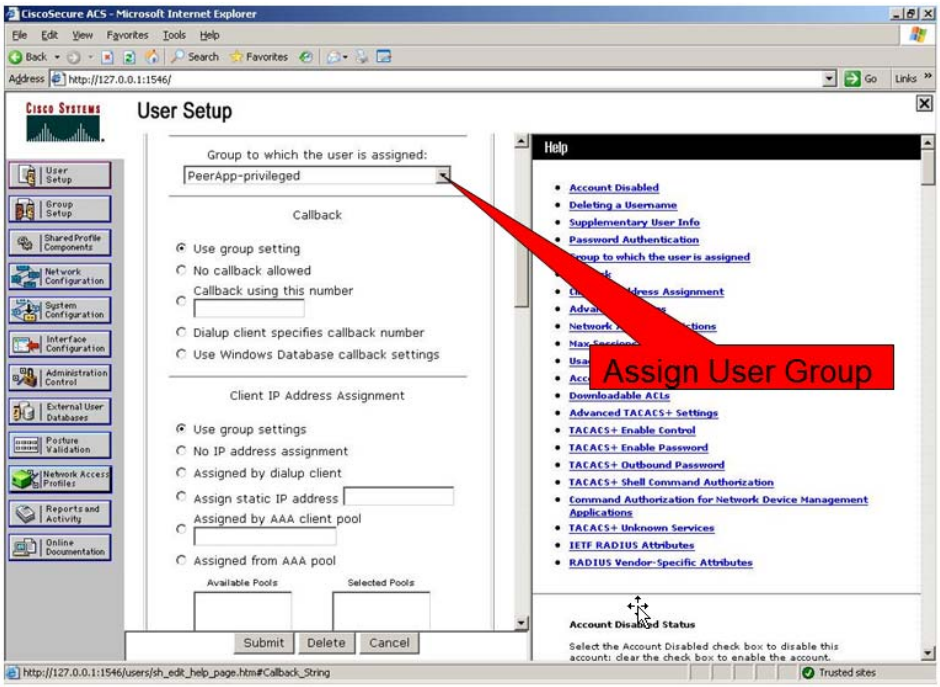
Figure 3-13 Cisco Secure ACS Group Setup: VDSTC-privileged



Step 4 Configure the VDS TC users on the TACACS+ server and configure the following settings:

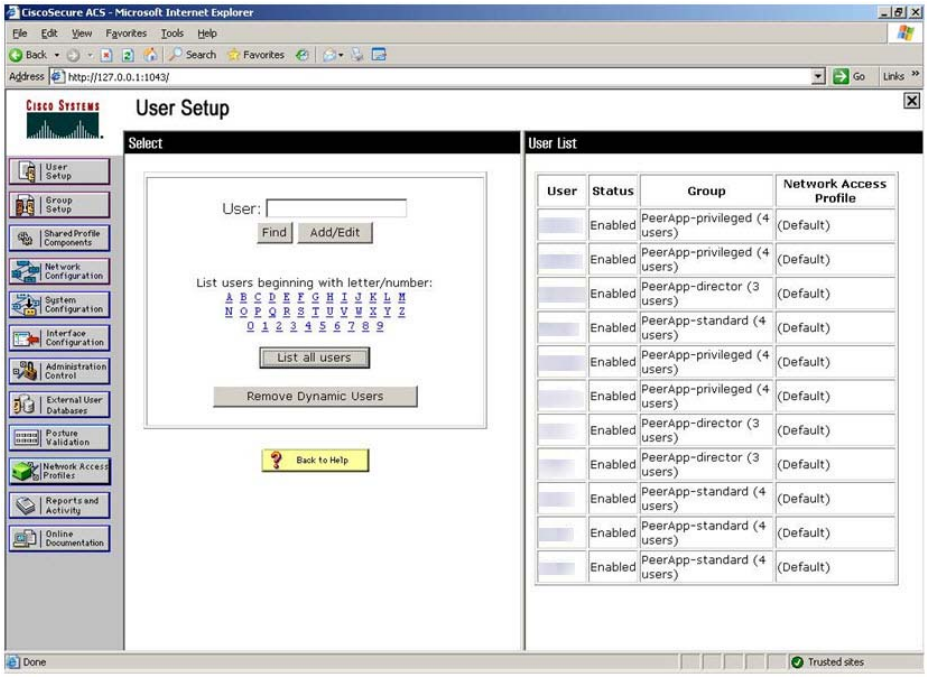
- In the User Setup section in the CiscoSecure PAP area, enter a password in the Password and Confirm Password text boxes. Currently only clear passwords are supported.
- Choose a group to assign to the user from the Group to Which User is Assigned drop-down list box. Assign the user to either the VDSTC-standard, VDSTC-director, or VDSTC-privileged group depending on the privileges they should have.
- In the Advanced TACACS+ Settings section choose **Use Cisco secure PAP Password**.

Figure 3-14 User Setup



Step 5 Once users are associated with the correct privileges, you can view the groups the users are assigned to from the User List, as shown in the following figure:

Figure 3-15 Cisco Secure ACS User List

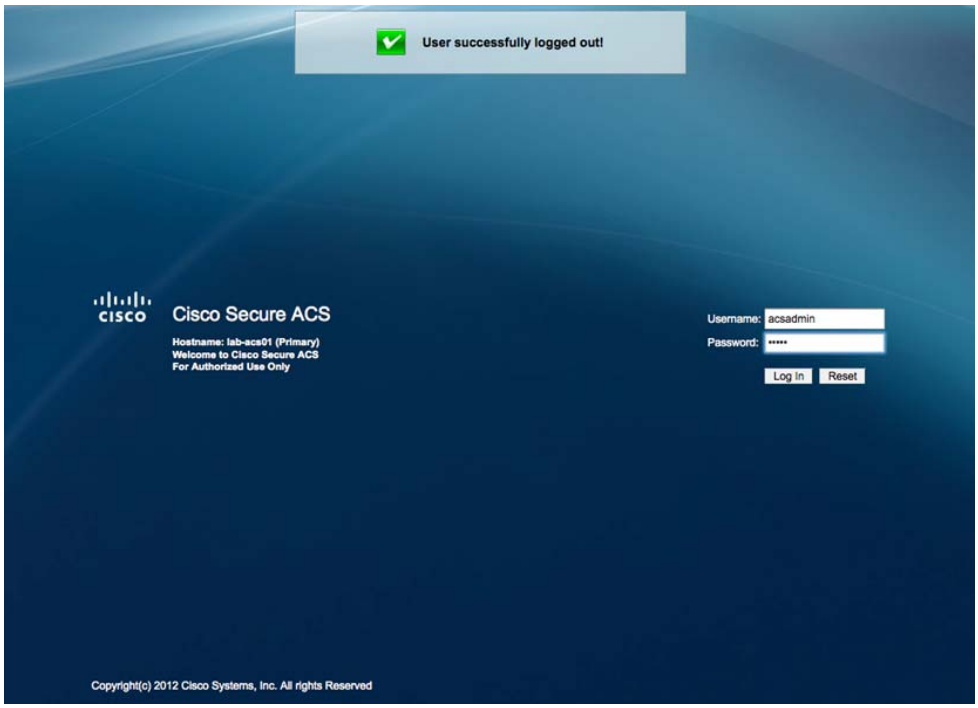


Configuring TACACS+ for VDS TC Support, Using Cisco Secure ACS Release 5.4

To configure TACAC+ support for the VDS TC platform on the Cisco Secure ACS server, perform the following steps on the Cisco Secure ACS server:

Step 1 Using HTTPS, log into the Cisco Secure ACS server. When prompted, enter your username and password for the server.

Figure 3-16 Cisco Secure 5.4 ACS Login Window



Step 2 From the Cisco Secure ACS main window, choose **Network Resources > Network Device Groups > Network Devices and AAA Clients**.

Step 3 Click Create and in the window that appears, enter the following information:

a. **Name:** Enter the name of the VDS TC system.

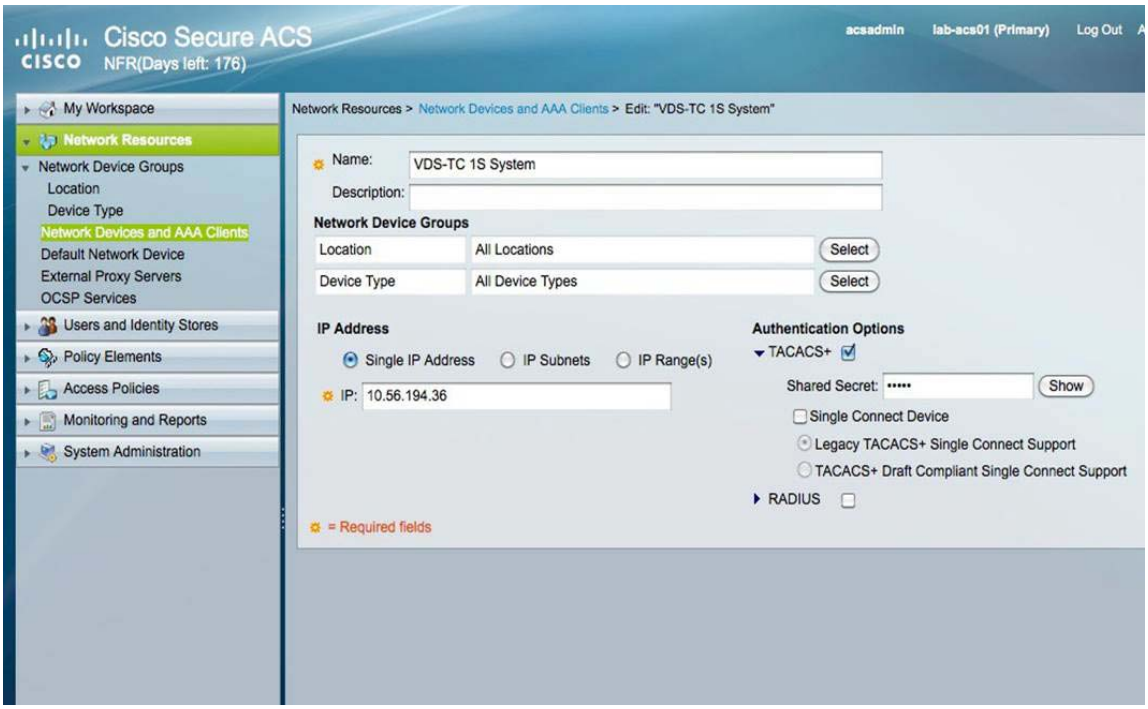
b. Authentication options:

- Check the **TACACS+** check box.
- In the Shared Secret field, enter the shared secret that corresponds to the shared secret that you will configure in the <tacacs_secret> tag in the cluster configuration file. See the Configuring the VDS-TC Management Server for TACACS+ for configuring the cluster configuration file for this value.

c. Click the **Single IP Address** radio button and in the IP address field, enter the IP address of the VDS TC management server.

Step 4 Click **Submit**. Your results should look similar to the following figure.

Figure 3-17 Create New AAA Client



Step 5 Next you will create three identity groups, one for Admins, one for directors, and one for the help desk team. To create these groups, choose **Users and Identity Stores > Identity Groups**.

Step 6 Click **Create**, enter the following values for the first group:

a. **Name:** TC Admins

b. **Parent:** All Groups (this is the default)

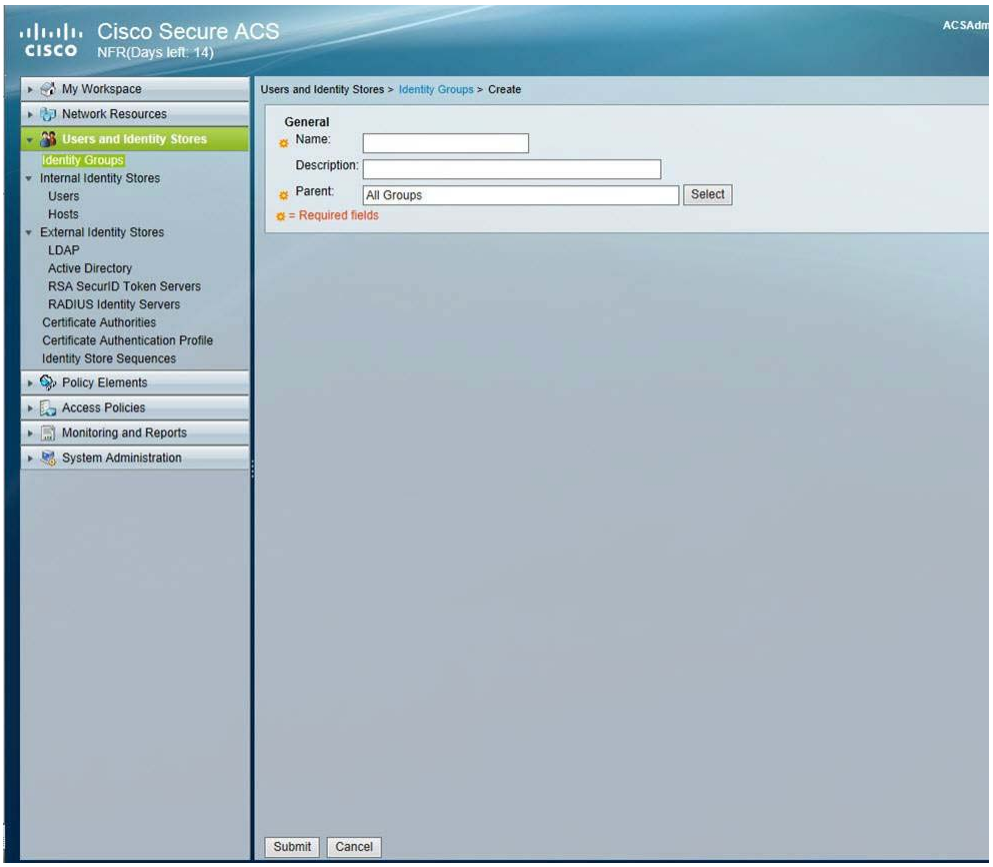
c. Click **Submit**.

Step 7 Repeat Step 6 to create the following groups:

- TC Director
- TC Help Desk

Note The group names can be any names that you would like to use, but the following configuration steps will use TC Admin, TC Director, and TC Help Desk as examples.

Figure 3-18 Create Identity Groups



Step 8 Next you will create three users, one for Admin access, one for Director access, and one for Help Desk access. To create these users, choose **Users and Identity Stores > Internal Identity Stores > Users**.

Step 9 Click **Create** to create the Admin user and complete the following information:

a. **Name:** Enter the name for the Admin user, for example tcadmin.

b. **Identity Group:** Choose the Identity Group that you created for the Admin group, which was All Groups: TC Admins in our example.

c. Click **Submit**.

Step 10 Click Create to create the Director user account:

a. **Name:** Enter the name for the Director user, for example tcdirector.

b. **Identity Group:** Choose the Identity Group that you created for the Director group, which was All Groups: TC Director in our example.

c. Click **Submit**.

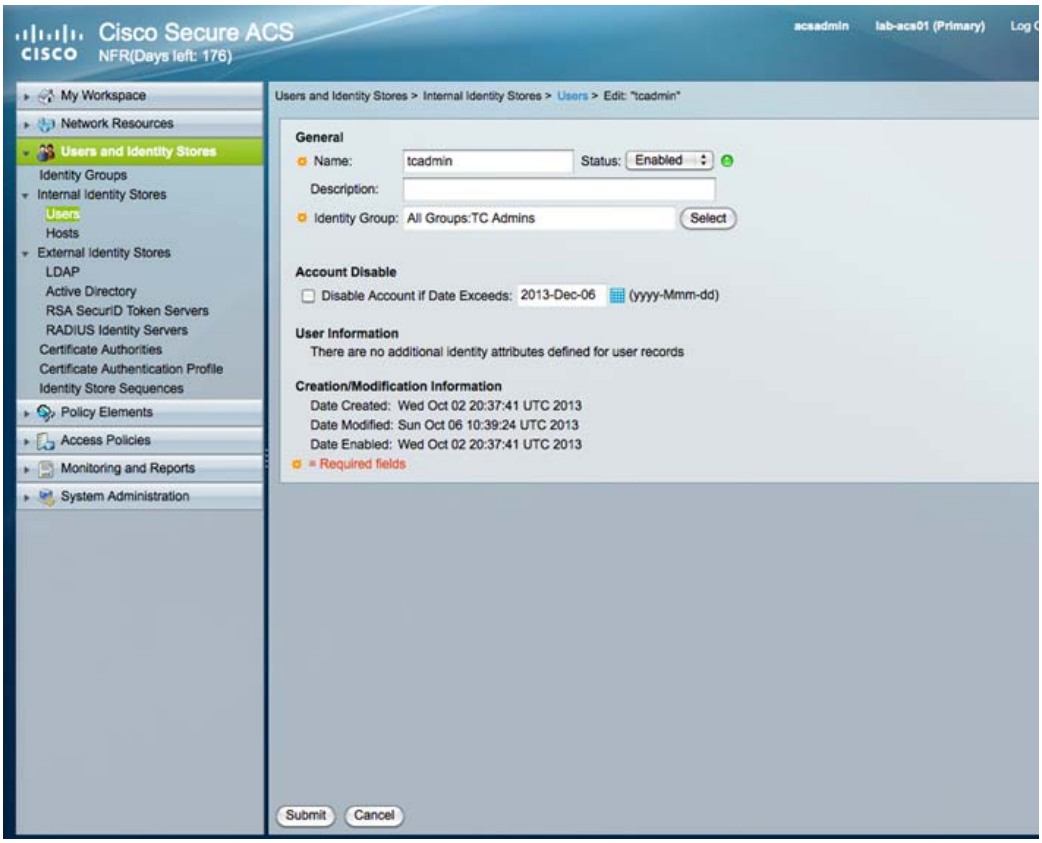
Step 11 Click Create to create the Help Desk user account:

a. **Name:** Enter the name for the Help Desk user, for example tchelp.

b. **Identity Group:** Choose the Identity Group that you created for the Help Desk group, which was All Groups: TC Help Desk in our example.

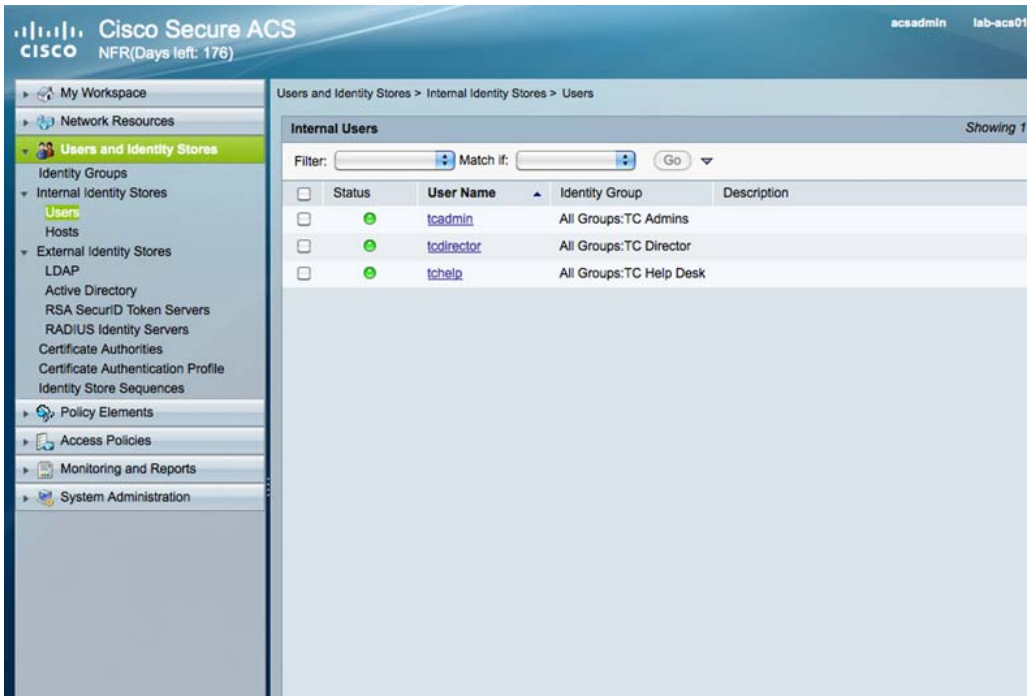
c. Click **Submit**.

Figure 3-19 Create User



Step 12 After you are done creating the users and groups, the output should look similar to the following:

Figure 3-20 Users List



Configure Policy Elements

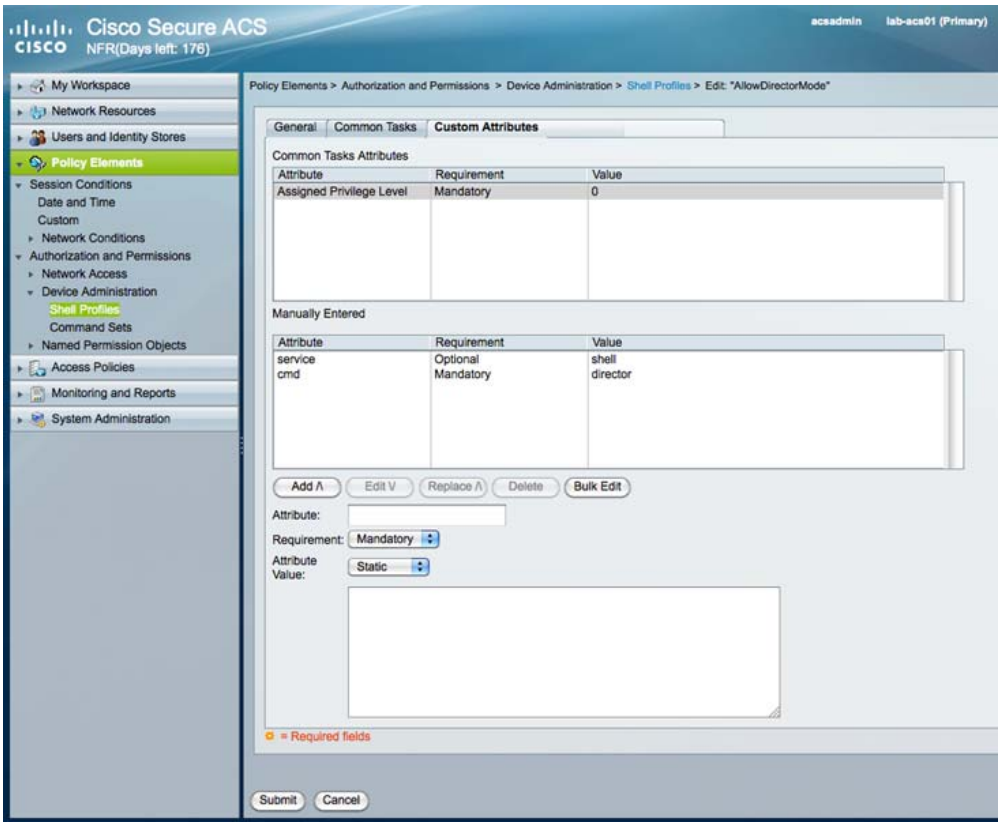
Follow these steps to create the policy elements in Cisco Secure ACS 5.4:

Step 1 Choose **Policy Elements > Authorizations and Permissions > Device Administration > Shell profiles**.

Step 2 Click **Create** and enter the following values to create a Director shell profile:

- On the General tab, in the Name field enter **AllowDirectorMode**.
- On the Common Tasks tab, configure the following:
  - Default Privilege: **Static** with a value of **0**
  - Maximum Privilege: **Not in use**
  - All Shell Attributes should be set to **Not in Use**.
- On the Custom Attributes tab add the following attributes:
  - attribute: service, requirement: optional, value: shell
  - attribute: cmd, requirement: mandatory, value: director

Figure 3-21 Director Custom Attributes



Step 3 Click **Submit**.

Step 4 Choose **Policy Elements > Authorizations and Permissions > Device Administration > Command Sets**.

Step 5 Click **Create** and enter the following values to create the AllowDirectorMode command sets:

- In the Name field, enter **AllowDirectorMode**.
- Make sure the "Permit any command that is not in the table below" check box is *unchecked*.
- Add the director command
  - Choose **Permit**.
  - In Command field enter **director**.
  - Leave Arguments field empty.
  - Click **Add**.
- Add the enable command
  - Choose **Permit**.
  - In Command field enter **enable**.
  - Leave Arguments field empty.
  - Click **Add**.
- Click **Submit**.

Step 6 Click **Create** and enter the following values to create the AllowEnableMode command sets:

- In the Name field, enter **AllowEnableMode**.
- Make sure the "Permit any command that is not in the table below" check box is *unchecked*.
- Add the enable command
  - Choose **Permit**.
  - In Command field enter **enable**.
  - Leave Arguments field empty.
  - Click **Add**.
- Click **Submit**.

Step 7 Click **Create** and enter the following values to create the DenyEnableMode command sets:

- In the Name field, enter **DenyEnableMode**.
- Make sure the "Permit any command that is not in the table below" check box is *unchecked*.
- Add the enable command
  - Choose **Permit**.
  - In Command field enter **show**.
  - Leave Arguments field empty.
  - Click **Add**.
- Click **Submit**.

Configure Access Policies

Follow these steps to configure the access polices:

Step 1 Choose **Access Policies > Access Services > Default Device Admin**.

Step 2 Under Policy Structure, check the **Identity** and **Authorization** check boxes.

Step 3 Click **Submit**.

Step 4 From the Allowed Protocols tab, check only the **Allow PAP/ASCII** check box.

Step 5 Click **Submit**.

Step 6 Choose **Access Policies > Access Services > Default Device Admin > Identity**.

Step 7 On the window that appears, click the **Single Result Selection** radio button.

Step 8 In the Advanced Options section, make sure the following settings are configured:

- If authentication failed: **Reject**
- If user not found: **Reject**
- If process failed: **Drop**

Step 9 If you made any changes to the Access Policies > Access Services > Default Device Admin > Identity window, click **Save Changes**.

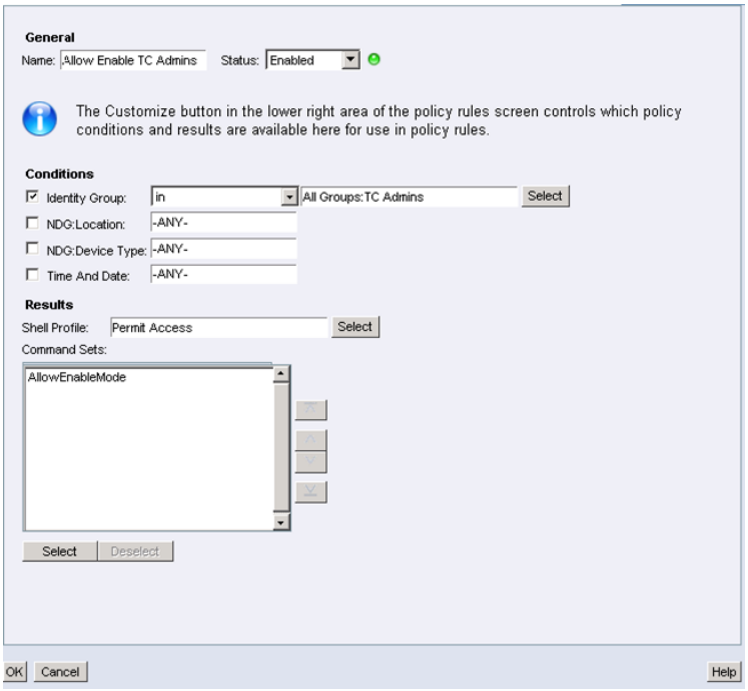
Step 10 Choose **Access Policies > Access Services > Default Device Admin > Authorization**.

Step 11 Click **Create** to create a rule. In the window that appears, enter the following values:

- Name: **Allow Enable TC Admins**
- Status: **Enabled**

- Identity Groups: **Check** the check box, choose **in** from the drop-down list and click **Select** and choose the group that you created for the Admins group. In our example, this is TC Admins.
- Shell Profile: Select **Permit Access**
- Commands Sets: Select **AllowEnableMode**
- Click **OK**.

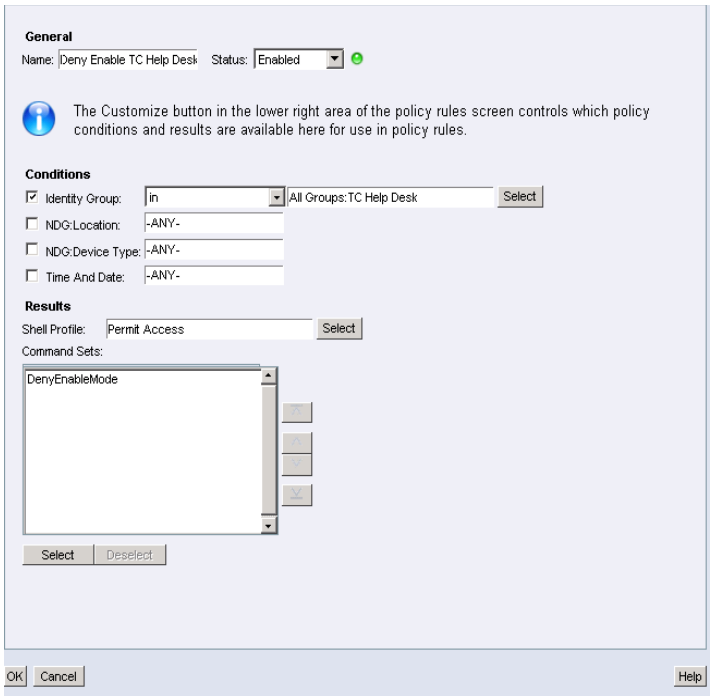
Figure 3-22 Allow Enable TC Admins Settings



Step 12 Click **Create** to create a rule. In the window that appears, enter the following values:

- Name: **Deny Enable TC Help Desk**
- Status: **Enabled**
- Identity Groups: **Check** the check box, choose **in** from the drop-down list and click **Select** and choose the group that you created for the Help Desk group. In our example, this is TC Help Desk.
- Shell Profile: Select **Permit Access**
- Commands Sets: Select **DenyEnableMode**
- Click **OK**.

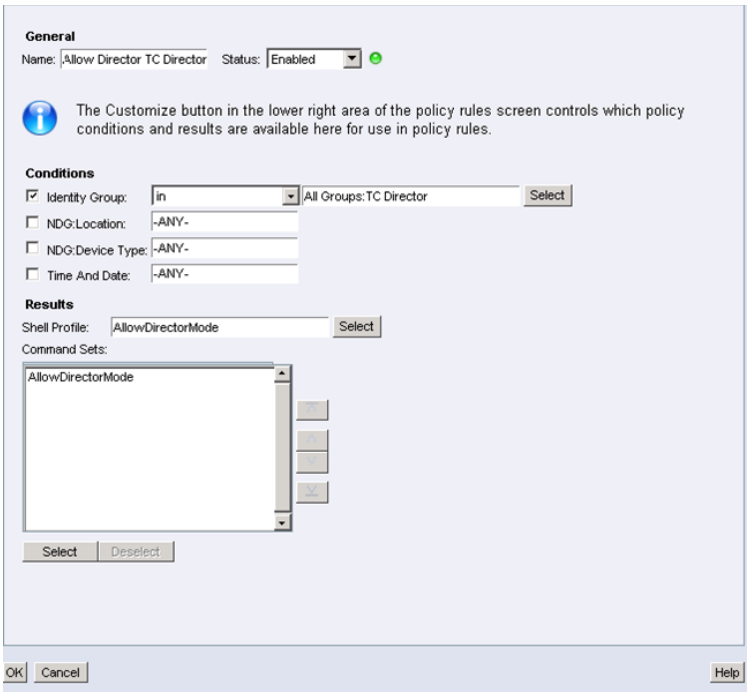
Figure 3-23 Deny Enable TC Help Desk Settings



Step 13 Click **Create** to create a rule. In the window that appears, enter the following values:

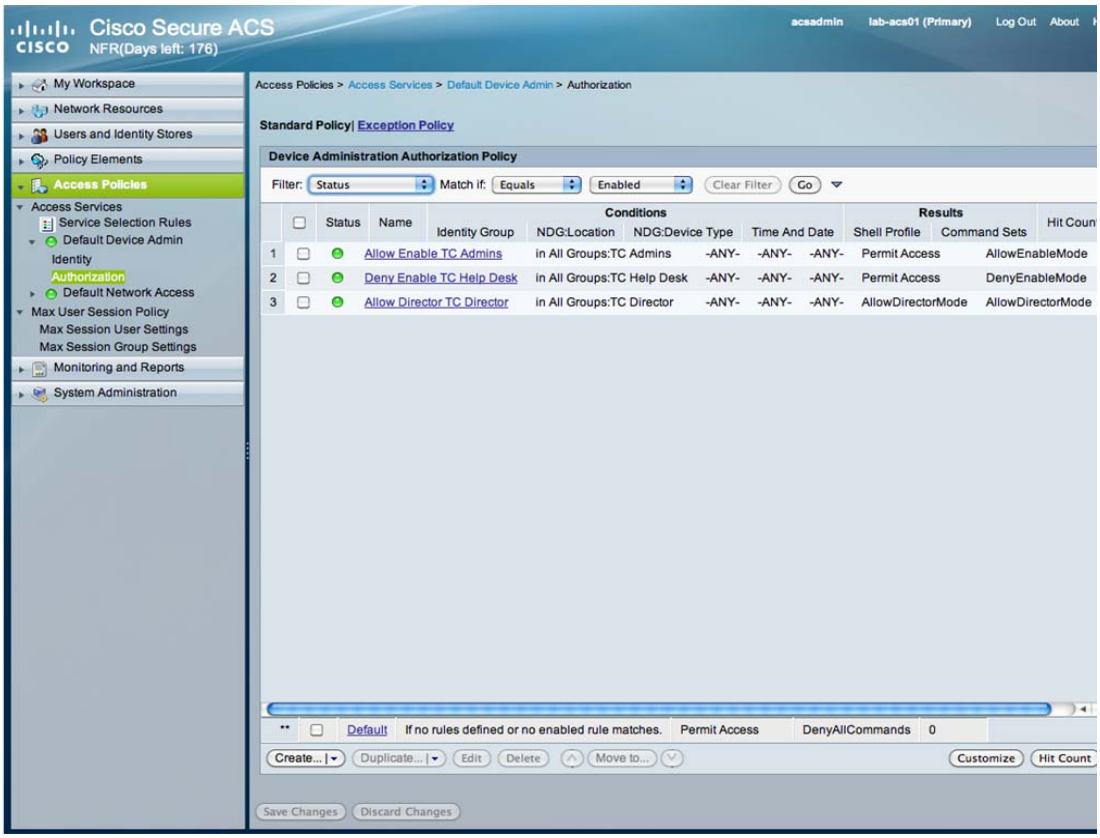
- Name: **Allow Director TC Director**
- Status: **Enabled**
- Identity Groups: **Check** the check box, choose **in** from the drop-down list and click **Select** and choose the group that you created for the Director group. In our example, this is TC Director.
- Shell Profile: Select **AllowDirectorMode**
- Commands Sets: Select **AllowDirectorMode**
- Click **OK**.

Figure 3-24 Allow Director TC Director Settings



Step 14 When you are done creating the authorization polices, the window should look like the following:

Figure 3-25 Authorization Settings



Configuring the VDS-TC Management Server for TACACS+

Configure and license the system (*see* the *Cisco Videoscape Distribution Suite Transparent Caching Installation Guide* [part number OL-28015-02]) and add the following configuration to the <mgmt-config> section in the VDS TC configuration file, where *server_ip* is the IP address of the TACACS+ server and *secret* is the shared secret that is configured on the TACACS+ server for the VDS TC platform:

```
<TACACS_configuration>
    <tacacs_server_ip>server_ip</tacacs_server_ip>
    <tacacs_secret>secret</tacacs_secret>
</TACACS_configuration>
```

For example:

```
<TACACS_configuration>
    <tacacs_server_ip>10.1.1.65</tacacs_server_ip>
    <tacacs_secret>SECRET-1234</tacacs_secret>
</TACACS_configuration>
```

Note The TACACS+ secret is a string that is configured on both sides and must match on the TACACS+ Server and the VDS TC management server.

After the TACACS+ configuration is applied, TACACS+ users can log into the VDS TC CLI using Telnet or SSH, or log into the VDS TC Manager.

Note Any changes that you make to the VDS TC configuration file will only take affect after you import the configuration file into the VDS TC management server. For information on how to import this file, see Working with the Configuration Files in Chapter 2, "Working with Cisco VDS TC Management Tools".

Using CLI Commands

The VDS TC platform is controlled using a set of CLI commands, allowing full control over its operational modes. The CLI commands are divided into three categories:

- **Regular mode commands:** From this mode you can display version and licensing information for the VDS TC platform, and you can access and manipulate the system log. You cannot perform system configuration changes from this mode.
- **Enable mode commands:** From this mode you have full control over the system configuration, cache content manipulation, networking behavior, licensing, platform operation state, and you can manage the software version on which the system runs.
- **Configuration mode commands:** From this mode you can make platform configuration changes. Changes you make are stored but are implemented only when you use the **apply** command.

To access the VDS TC CLI use the username **admin.** The default password is the serial number of the chassis.

The following is a sample authentication session:

```
Login as: admin
Using keyboard-interactive authentication.
Password:
Cli version - 5.7.3b54
Snmp version - VDS-TC Transparent Caching mgmt software version 5.7.3b54
console>
```

Critical user activities are recorded in the system eventlog, allowing administrators to monitor system activity.

This chapter provides a reference for all of the CLI commands that are available for an Integrated Appliance solution. *See* Chapter 6, "CLI Reference" for a full list of available CLI commands.

Regular Mode Commands

This section describes the commands that are available in Regular mode. These commands are also available in Enable mode. The Regular Mode Commands table lists the commands that are described in this section.

Note All of the CLI commands work with both IPv4 and IPv6.

Note All commands and their parameters are case sensitive.

Table 4-1 Regular Mode Commands

Command	Description
arp	Displays the ARP table
current_cli_users	Displays the list of VDS TC admin users who are currently logged into the VDS TC CLI
direction	Calculates the visible subnets on the interface
dmesg	Displays the message buffer
enable	Enters Enable mode
eventlog	Provides access to event log operations
exit	Exits the current mode
help	Displays the list of available commands for the mode you are in when you execute the command
ifconfig	Displays the interface(s)
iostat	Displays extended I/O statistics
jumbo	Sends jumbo echo messages
ping	Sends echo messages
show	Displays run-time information
traceroute	Displays the route used by the packet to reach its destination

arp

To display the ARP table, use the **arp** command.

arp

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **arp** command:

```
console> arp
Address      HWtype        HWaddress          Flags Mask  Iface
192.168.0.2  ether         00:17:65:C7:10:42   C          eth0
```

current_cli_users

To display the admin users who are currently logged on to the VDS TC CLI, use the **current_cli_users** command.

current_cli_users

Note You must be logged on using an admin username and password to view the output from this command. This command only displays admin users that are logged directly into the VDS TC CLI. It does *not* display users that have used the **sudo** command from the Linux CLI to access the VDS TC CLI.

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following example displays a list of users currently logged on to the system:

```
console> current_cli_users
admin    pts/1        Apr  4 21:32 (10.21.150.101)
```

direction

To calculate the visible subnets on the specified interface, use the **direction** command.

direction *interface_name*

Syntax Description

interface_name	The interface for which you want to display the subnets, for example eth0.

Command Modes

Regular mode and Enable mode

Examples

The following sample displays the visible subnets on the interface eth0 using the **direction** command:

```
console> direction eth0
tcpdump: WARNING: eth0: no IPv4 address assigned
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
1000 packets captured
1006 packets received by filter
0 packets dropped by kernel
    453 10.11.12.0/24
    479 10.11.18.0/24
      6 10.11.80.0/24
     59 10.56.10.0/24
      1 64.101.47.0/24
```

```
      2 8.8.8.0/24
console>
```

dmesg

To displays the message buffer of the kernel, use the **dmesg** command.

dmesg

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **dmesg** command:

```
console> dmesg
Ending clean XFS mount for filesystem: sdl2
XFS mounting filesystem sdm3
Ending clean XFS mount for filesystem: sdm3
XFS mounting filesystem sdm2
Ending clean XFS mount for filesystem: sdm2
XFS mounting filesystem sdk3
Ending clean XFS mount for filesystem: sdk3
XFS mounting filesystem sdk2
Ending clean XFS mount for filesystem: sdk2
XFS mounting filesystem sdo3
Ending clean XFS mount for filesystem: sdo3
FS mounting filesystem sdo2
Ending clean XFS mount for filesystem: sdo2
XFS mounting filesystem sdn3
Ending clean XFS mount for filesystem: sdn3
XFS mounting filesystem sdn2
Ending clean XFS mount for filesystem: sdn2
monitor[19319]: segfault at 1500 ip 00007fcc02938d9b sp 00007fcbff400040 error 4 in
ld-2.9.so[7fcc02930000+1e000]
device eth0 entered promiscuous mode
device eth0 left promiscuous mode
device eth0 entered promiscuous mode
device eth0 left promiscuous mode
```

enable

Enable mode allows you to access CLI commands and make configuration changes. These changes include cache content manipulation, networking behavior, licensing, and managing the software version on which the system runs. To enter enable mode, use the **enable** command.

enable

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode

Examples

The following example shows the use of the **enable** command. After the user enters the correct password, the system enters Enable mode, as indicated by the pound sign (#):

```
console> enable
Password:
console#
```

eventlog

The event log lists all of the log messages sent to the system log by all of the VDS TC service components (applications, CLI and SNMP). To view or export the content of the event log, use the **eventlog** command.

eventlog {**date** *date*| **export** *tftp_server filename* | **show** | **tail**}

Note

Additional parameters for this command are available in Enable mode. Refer to eventlog, page 4-24 for a description of the Enable mode parameters.

Syntax Description

date *date*	Displays the eventlog for the date specified with the *date* parameter. The *date* parameter is in the format of DD-MM-YYYY.
export *tftp_server filename*	Exports the content of the event log to the TFTP server with the name or IP address specified with the *tftp_server* parameter.
show	Displays the event log entries.
tail	Displays the online event log entries.

Command Modes

Regular mode and Enable mode

Examples

The following example exports the eventlog to a TFTP server with an IP address of 192.168.77.14 with a filename of eventlog-exported:

```
console> eventlog export 192.168.77.14 eventlog-exported
```

The following is sample output from the **eventlog show** command:

```
console> eventlog show
Aug 18 12:29:07 ce-1 pang[21134]: /mnt/vol9        mounted          active    ce-1
678        401        276        40.80
Aug 18 12:29:07 ce-1 pang[21134]: /mnt/vol10       mounted          active    ce-1
678        417        261        38.48
Aug 18 12:29:07 ce-1 pang[21134]: /mnt/vol11       mounted          active    ce-1
678        457        220        32.55
Aug 18 12:29:07 ce-1 pang[21134]: /mnt/vol12       mounted          active    ce-1
678        397        281        41.43
```

```
Aug 18 12:29:07 ce-1 pang[21134]: /mnt/vol13        mounted          active    ce-1
678        413        265        39.06
Aug 18 12:29:07 ce-1 pang[21134]: /mnt/vol14        mounted          active    ce-1
678        403        274        40.52
Aug 18 12:29:07 ce-1 pang[21134]: /mnt/vol15        mounted          active    ce-1
678        401        277        40.84
Aug 18 12:36:06 ce-1 pang[21134]: volume             state      availability owner
total      free       used       usage
Aug 18 12:36:06 ce-1 pang[21134]: /mnt/vol1         mounted          active    ce-1
678        403        274        40.50
```

Note In a Cisco VDS TC installation that uses the Cisco Blade Servers, you may see "cluster has been enabled" followed by "cluster has been degraded" SNMP messages in the eventlog. When the Cisco VDS TC system does not receive traffic on the cache engine interfaces, it believes there may be a problem with the interfaces. In an attempt to "fix" this perceived problem, the system disables and enables the application, causing the "cluster has been enabled" and the "cluster has been degraded" messages to appear in the logs.

The following is sample output from the **eventlog tail** command:

```
console> eventlog tail
Aug 18 12:50:05 ce-1 pang[21134]: /mnt/vol6         mounted          active    ce-1
678        399        278        41.09
Aug 18 12:50:05 ce-1 pang[21134]: /mnt/vol7         mounted          active    ce-1
678        394        283        41.82
Aug 18 12:50:05 ce-1 pang[21134]: /mnt/vol8         mounted          active    ce-1
678        416        261        38.54
Aug 18 12:50:05 ce-1 pang[21134]: /mnt/vol9         mounted          active    ce-1
678        401        276        40.81
Aug 18 12:50:05 ce-1 pang[21134]: /mnt/vol10        mounted          active    ce-1
678        417        261        38.50
Aug 18 12:50:05 ce-1 pang[21134]: /mnt/vol11        mounted          active    ce-1
678        457        220        32.57
Aug 18 12:50:05 ce-1 pang[21134]: /mnt/vol12        mounted          active    ce-1
678        397        281        41.44
```

exit

To exit any mode or close an active CLI session use the **exit** command. In Enable mode, this command returns the user to Regular mode. In Regular mode, this command terminates the session and the user is logged out of the CLI session.

exit

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode, Enable mode, and Configuration mode

Examples

The following example shows how to exit the current session:

```
console> exit
```

help

To display the CLI commands that are available in the current mode, with a short description of each command, use the **help** command. For example, if you enter the **help** command in Regular mode, then the command displays only the commands that are available in Regular mode with a short description of each command.

help

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode, Enable mode, and Configuration mode

Examples

The following example displays the list of CLI commands that are available in Regular mode:

```
console> ?
arp                  Show arp table
current_cli_users  Show currently logged in cli users
direction            Calculate seen subnets on interface
dmesg                Display dmesg
enable               Enter privileged mode
eventlog             Event log commands
exit                 Exit current mode
help                 Commands description
ifconfig             Display interface(s)
iostat               Display IO statistics
jumbo                Send jumbo echo messages
ping                 Send echo messages
show                 Show run-time information
traceroute           Print the route packets take to network host
```

ifconfig

To display the details of the interfaces, use the **ifconfig** command.

ifconfig

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **ifconfig** command:

```
console> ifconfig
bond0     Link encap:Ethernet  HWaddr D4:8C:B5:4D:C0:14
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:37161009 errors:0 dropped:0 overruns:0 frame:0
          TX packets:953496 errors:0 dropped:0 overruns:0 carrier:0
```

```
          collisions:0 txqueuelen:0
          RX bytes:3045286713 (2904.2 Mb)  TX bytes:245299515 (233.9 Mb)

bond0:1   Link encap:Ethernet  HWaddr D4:8C:B5:4D:C0:14
          inet addr:10.56.194.36  Bcast:10.56.195.255  Mask:255.255.254.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1

eth0      Link encap:Ethernet  HWaddr D4:8C:B5:4D:C0:14
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:19175744 errors:0 dropped:0 overruns:0 frame:0
          TX packets:953496 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1907814608 (1819.4 Mb)  TX bytes:245299515 (233.9 Mb)
eth1      Link encap:Ethernet  HWaddr D4:8C:B5:4D:C0:14
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:17985265 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1137472105 (1084.7 Mb)  TX bytes:0 (0.0 b)

        <output omitted>
```

iostat

To report CPU statistics and input/output statistics for devices and partitions, use the **iostat** command.

iostat [**-t** *interval* [**-k** *count*]]

Syntax Description

-t *interval*	The amount of time, in seconds between each report. The default is 5 seconds.
-k *count*	Used in conjunction with the interval parameter. If the count parameter is specified, the count determines the number of reports generated at the specified interval. If the interval parameter is specified without the count parameter, the **iostat** command generates reports continuously until you press **Ctrl-C**.

Command Modes

Regular mode and Enable mode

Examples

The following example generates two I/O statistics reports two seconds apart:

```
console> iostat -t 2 -k 2
Linux 2.6.27.7-llpf-9-default (ce-1)    08/18/09        _x86_64_

Time: 12:57:39
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          13.32    0.00   17.96    5.03    0.00   66.93

Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda               0.01     0.00   12.78    1.08   802.07    34.63   120.67     0.15
10.80   4.95   6.86
```

```
sda1              0.00     0.00    0.00    0.00     0.00     0.00    39.76     0.00
4.84   3.40   0.00
sda2              0.00     0.00    0.00    0.00     0.01     0.00    10.66     0.00
1.48   1.36   0.00
sda3              0.01     0.00    0.24    0.58     3.96     2.42    15.55     0.00
4.42   3.93   0.32
sda4              0.00     0.00   12.54    0.51   798.10    32.21   127.30     0.15
11.20   5.23   6.82
sdb               0.01     0.00  111.63    1.06  7095.08    32.91   126.51     2.67
23.68   5.73  64.53
sdb1              0.00     0.00    0.00    0.00     0.00     0.00    42.42     0.00
4.97   2.90   0.00
sdb2              0.00     0.00    0.00    0.00     0.01     0.00    13.85     0.00
1.60   1.32   0.00
sdb3              0.00     0.00    0.46    0.58     2.20     2.41     8.86     0.02
15.72  12.90   1.34
sdb4              0.01     0.00  111.16    0.48  7092.86    30.50   127.61     2.65
23.76   5.77  64.45
sdc               0.01     0.00   48.38    1.03  3045.81    33.49   124.64     0.23
4.65   3.41  16.87
<output omitted>

Time: 12:57:41
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          15.94    0.00   24.26    5.15    0.00   59.33

Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda               0.00     0.00   64.50    7.50  4084.75   224.00   119.69     0.46
6.42   4.75  34.20
sda1              0.00     0.00    0.00    0.00     0.00     0.00     0.00     0.00
0.00   0.00   0.00
sda2              0.00     0.00    0.00    0.00     0.00     0.00     0.00     0.00
0.00   0.00   0.00
sda3              0.00     0.00    0.00    4.00     0.00    14.25     7.12     0.00
1.00   1.00   0.40
sda4              0.00     0.00   64.50    3.50  4084.75   209.75   126.31     0.46
6.74   5.03  34.20
sdb               0.00     0.00  115.00   15.50  7341.50   510.25   120.33     2.05
15.36   5.53  72.20
sdb1              0.00     0.00    0.00    0.00     0.00     0.00     0.00     0.00
0.00   0.00   0.00
sdb2              0.00     0.00    0.00    0.00     0.00     0.00     0.00     0.00
0.00   0.00   0.00
sdb3              0.00     0.00    0.00    8.00     0.00    30.25     7.56     0.10
13.00  13.00  10.40
<output omitted>
```

jumbo

To send jumbo echo messages, use the **iostat** command.

jumbo [**-c** *counter*] [**-I** {*IP_address* | *interface*}] *destination*

Syntax Description

-c *counter*	The number of times the request is generated.
-I {*IP_address* | *interface*}	The interface IP address or interface name from which the echo requests are sent.
destination	The destination to which the jumbo echo message will be sent.

Command Modes

Regular mode and Enable mode

Examples

The following example sends jumbo echo messages to 192.168.0.202 sourced from the eth0 interface:

```
console> jumbo -I eth0 192.168.0.202
PING 192.168.0.202 (192.168.0.202) from 192.168.0.202 eth0: 8972(9000) bytes of data.
8980 bytes from 192.168.0.202: icmp_seq=1 ttl=64 time=0.041 ms
8980 bytes from 192.168.0.202: icmp_seq=2 ttl=64 time=0.049 ms
8980 bytes from 192.168.0.202: icmp_seq=3 ttl=64 time=0.032 ms
8980 bytes from 192.168.0.202: icmp_seq=4 ttl=64 time=0.033 ms
8980 bytes from 192.168.0.202: icmp_seq=5 ttl=64 time=0.030 ms

--- 192.168.0.202 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3997ms
rtt min/avg/max/mdev = 0.030/0.037/0.049/0.007 ms
```

The following example sends two jumbo echo messages to 192.168.0.202 sourced from the eth0 interface:

```
console> jumbo -c 2 -I eth0 192.168.0.202
PING 192.168.0.202 (192.168.0.202) from 192.168.0.202 eth0: 8972(9000) bytes of data.
8980 bytes from 192.168.0.202: icmp_seq=1 ttl=64 time=0.041 ms
8980 bytes from 192.168.0.202: icmp_seq=2 ttl=64 time=0.049 ms

--- 192.168.0.202 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3997ms
rtt min/avg/max/mdev = 0.030/0.037/0.049/0.007 ms
```

ping

To diagnose basic network connectivity, use the **ping** command. The **ping** command uses the ICMP protocol mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway, and displays the round-trip time for the echo response to arrive back to the server on which the command was issued.

ping [**-c** *counter*] [**-I** {*IP_address* | *interface*}] *destination*

Syntax Description

-c *counter*	The number of ICMP echo requests to be sent to the destination address. If you do not specify the number of echo requests to send, the ping will continue until you press **Ctrl-C**.

-I {*IP_address* | *interface*}	The interface IP address or interface name from which the pings are sent.
destination	The destination to which the ping messages will be sent. This value can be an IP address or a hostname.

Command Modes

Regular mode and Enable mode

Usage Guidelines

To abort the ping command, press **Ctrl-C**.

Examples

The following example pings the destination at 192.168.0.202 sourced from the eth0 interface until Ctrl-C is pressed:

```
console> ping -I eth0 192.168.0.202
PING 192.168.0.202 (192.168.0.202) from 192.168.0.202 eth0: 56(84) bytes of data.
64 bytes from 192.168.0.202: icmp_seq=1 ttl=64 time=0.023 ms
64 bytes from 192.168.0.202: icmp_seq=2 ttl=64 time=0.035 ms
64 bytes from 192.168.0.202: icmp_seq=3 ttl=64 time=0.028 ms
64 bytes from 192.168.0.202: icmp_seq=4 ttl=64 time=0.020 ms
64 bytes from 192.168.0.202: icmp_seq=5 ttl=64 time=0.029 ms
64 bytes from 192.168.0.202: icmp_seq=6 ttl=64 time=0.030 ms
64 bytes from 192.168.0.202: icmp_seq=7 ttl=64 time=0.032 ms
64 bytes from 192.168.0.202: icmp_seq=8 ttl=64 time=0.029 ms
64 bytes from 192.168.0.202: icmp_seq=9 ttl=64 time=0.030 ms
64 bytes from 192.168.0.202: icmp_seq=10 ttl=64 time=0.030 ms
--- 192.168.0.202 ping statistics ---
10 packets transmitted, 0 received, 0% packet loss, time 19023ms
rtt min/avg/max/mdev = 0.019/0.027/0.037/0.006 ms
```

show

To display run-time information related to the operational environment of VDS TC, use the **show** command.

show {**detection_rules | dstat | eth_status | detection_rules | eventlog | leader | process | status | systemid | time | uptime | version | vip | volumes**}

Syntax Description

detection_rules	Display detection rules
dstat	Display IO statistics
eth_status	Displays the eth status.
detection_rules	Displays detection rules.
eventlog	Displays the platform event log. This command displays the same results as the **eventlog show** command.
leader	Displays the hostname of the current cluster leader. The cluster leader manages resources used by VDS TC.

process	Displays the status of VDS TC components (caching application, spread, and apache) as they run on the platform. The output of this command is relevant for maintenance engineers. An equivalent Server mode command is available for each of the servers that are part of the VDS TC cluster: **process_server.**
status	Displays the cluster administrative and application status.
systemid	Displays the system serial number.
time	Displays the system date and time.
uptime	Displays the uptime for the appliance.
version	Displays the software version.
vip	Show VIP information
volumes	Displays the mounted volumes.

Note Additional parameters for this command are available in Enable mode. Refer to the **show** command in the Enable Mode Commands section for a description of these parameters.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **show eth_status** command:

```
console> show eth_status
Blade     eth0    eth1    eth2    eth3    eth4    eth5    eth6    eth7
ce-1       UP      UP      UP     DOWN     UP      UP      UP      UP
```

The following is sample output from the **show eventlog** command:

```
console> show eventlog
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol9        mounted          active   ce-1
678        546        131        19.37
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol10       mounted          active   ce-1
678        548        129        19.15
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol11       mounted          active   ce-1
678        546        132        19.51
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol12       mounted          active   ce-1
678        546        131        19.44
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol13       mounted          active   ce-1
678        546        132        19.49
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol14       mounted          active   ce-1
678        544        134        19.76
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol15       mounted          active   ce-1
678        543        135        19.94
Apr 22 10:06:59 ce-1 pang[29533]: volume             state      availability owner
total      free       used       usage
<output omitted>
```

The following is sample output from the **show leader** command:

```
console> show leader
ce-1
```

The following is sample output from the **show process** command:

```
console> show process
admin     2243     1  0 Aug13 ?        00:00:00 -pang_cli
admin     3379  3378  0 12:58 pts/0    00:00:00 -pang_cli
admin     3865  3864  0 10:31 pts/1    00:00:00 -pang_cli
root      5762     1  0 Aug13 ?        00:00:06 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun    6192  5762  0 Aug16 ?        00:00:08 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
spread    6375     1  0 Aug13 ?        00:28:53 /usr/bin/spread -n ce-1 -c
/etc/spread.conf
wwwrun    8720  5762  0 Aug16 ?        00:00:08 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun    8993  5762  0 Aug17 ?        00:00:04 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun    9006  5762  0 Aug17 ?        00:00:04 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun    9007  5762  0 Aug17 ?        00:00:04 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun   11867  5762  0 Aug16 ?        00:00:07 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun   13669  5762  0 Aug16 ?        00:00:08 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
root     19290     1  0 Aug17 ?        00:00:00 /opt/pang/mgmt/avalon/sbin/snmptrapd -f
-Osq -Ls user -c /opt/pang/mgmt/avalon/sbin/snmptrapd.conf 127.0.0.1
root     20933     1  0 Aug17 ?        00:04:31 /opt/pang/mgmt/avalon/sbin/snmpd -f -A -LF
e /opt/pang/mgmt/avalon/var/log/snmpd.log -LS c u 192.168.0.202
root     20984     1  2 Aug17 ?        00:31:32 /opt/pang/cache/avalon/sbin/snmpd -f -A
-LF e /opt/pang/cache/avalon/var/log/snmpd.log -LS c u 127.0.0.1:10161
root     21134     1 99 Aug17 ?        3-13:27:17 /opt/pang/bin/pang -d -f
/opt/pang/conf/pang.conf
wwwrun   27907  5762  0 10:25 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
root     30012     1  0 Aug17 ?        00:03:49 /opt/pang/mgmt/bin/monitor -i lo
wwwrun   31197  5762  0 Aug16 ?        00:00:07 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun   31306  5762  0 Aug16 ?        00:00:07 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
```

The following is sample output from the **show status** command:

```
Console> show status
Operational state    Device state         Administrative state
 enabled              started              unlocked
```

The following is sample output from the **show systemid** command:

```
console> show systemid
*8GB9C4J
```

The following is sample output from the **show time** command:

```
console> show time
Mon Mar 31 2014 20:49:25 GMT+0200
```

The following is sample output from the **show uptime** command:

```
console> show uptime
49 days, 3h:07m:21s
```

The following is sample output from the **show version** command:

```
console> show version
VDS-TC Transparent Caching           cli version 5.7.3b54
management           VDS-TC Transparent Caching mgmt software version 5.7.3b54
PANG                 5.7.3b53         LLPF Version LLPF_05.7.3b53-54
```

The following is sample output from the **show volumes** command:

```
console> show volumes
Licensed volumes : 12
Volume name          State               Owner
/mnt/vol1            mounted             ce-1
/mnt/vol2            mounted             ce-1
/mnt/vol3            mounted             ce-1
/mnt/vol4            mounted             ce-1
/mnt/vol5            mounted             ce-1
/mnt/vol6            mounted             ce-1
/mnt/vol7            mounted             ce-1
/mnt/vol8            mounted             ce-1
/mnt/vol9            mounted             ce-1
/mnt/vol10           mounted             ce-1
/mnt/vol11           inactive
/mnt/vol12           inactive
```

traceroute

To discover the IP routes that packets will actually take when traveling to their destination, use the **traceroute** command. The **traceroute** command tracks the route of a packet across a TCP/IP network on its way to a given host. It utilizes the IP protocol time to live (TTL) field and attempts to elicit an ICMP TIME_EXCEEDED response from each gateway along the path to the destination.

traceroute [**-n**] *destination*

Syntax Description

-n	Forces the traceroute command to avoid mapping IP addresses to host names when displaying the output.
destination	The name or IP address of the destination.

Command Modes

Regular mode and Enable mode

Examples

The following example shows a traceroute to the destination 213.25.17.65:

```
console> traceroute 213.25.17.65
traceroute to 213.25.17.65 (213.25.17.65), 30 hops max, 40 byte packets
 1  192.168.0.2 (192.168.0.2)  1.316 ms   1.485 ms   1.292 ms
 2  10.1.2.253 (10.1.2.253)  0.656 ms   0.372 ms   0.747 ms
 3  212.150.66.65 (212.150.66.65)  1.481 ms   1.210 ms   1.044 ms
 4  212.150.6.137 (212.150.6.137)  5.517 ms   6.398 ms   6.574 ms
 5  gig0-1-gw1.thc.nv.net (212.143.200.33)  10.862 ms   10.842 ms   10.668 ms
 6  * coresw1-hfa-gw1.thc.nv.net (212.143.200.1)  6.721 ms   6.895 ms
 7  teng2-1-0.gw2.hfa.nv.net (212.143.8.196)  5.959 ms   9.117 ms   8.460 ms
 8  pos0-1-1.brdr2.lnd.nv.net (212.143.12.57)  77.423 ms *   75.177 ms
 9  GigabitEthernet3-1.GW1.LND9.ALTER.NET (146.188.55.61)  73.425 ms   69.143 ms 70.365 ms
10  * so-3-0-0.CR2.LND9.ALTER.NET (158.43.150.145)  157.824 ms   153.850 ms
11  so-0-3-0.XT2.LND2.ALTER.NET (146.188.15.245)  69.099 ms   69.834 ms *
12  GigabitEthernet1-0-0.BR1.LND18.ALTER.NET (146.188.4.42)  73.002 ms   72.350 ms
78.342 ms
```

```
13  GE.LND.opentransit.net (146.188.112.82)  70.164 ms   69.356 ms   69.688 ms
14  tengige0-8-0-0.lontr1.London.opentransit.net (193.251.129.81)  80.167 ms   80.134 ms
79.107 ms
15  xe-0-0-1-0.ffttr2.Frankfurt.opentransit.net (193.251.129.41)  92.546 ms   95.034 ms
94.282 ms
16  * * *
17  do.wro-ar3.z.wro-r1.tpnet.pl (213.25.5.154)  120.296 ms do.wro-ar3.z.wro-r2.tpnet.pl
(213.25.12.154)  134.187 ms   130.254 ms
18  * z-easy-com.wro-ar1.tpnet.pl (80.50.233.62)  236.345 ms   235.563 ms
19  z-easy-com.wro-ar1.tpnet.pl (80.50.233.62)  124.651 ms   125.124 ms   126.872 ms
```

The following example shows a traceroute to the destination 213.25.17.65 without resolving hostnames:

```
console> traceroute -n 213.25.17.65
traceroute to 213.25.17.65 (213.25.17.65), 30 hops max, 40 byte packets
 1  192.168.0.2 (192.168.0.2)  1.307 ms   2.133 ms   2.832 ms
 2  10.1.2.253 (10.1.2.253)  0.472 ms   0.477 ms   0.438 ms
 3  212.150.66.65 (212.150.66.65)  1.150 ms   1.391 ms   2.616 ms
 4  212.150.6.137 (212.150.6.137)  6.146 ms *   4.927 ms
 5  212.143.200.33 (212.143.200.33)  5.112 ms   5.374 ms   5.457 ms
 6  212.143.200.1 (212.143.200.1)  6.348 ms   7.459 ms   6.200 ms
 7  212.143.8.196 (212.143.8.196)  6.269 ms   6.208 ms   6.420 ms
 8  212.143.12.57 (212.143.12.57)  75.146 ms   74.670 ms   74.082 ms
 9  146.188.55.61 (146.188.55.61)  68.022 ms   68.586 ms   69.261 ms
10  158.43.150.145 (158.43.150.145)  76.673 ms *   76.009 ms
11  146.188.15.245 (146.188.15.245)  68.753 ms *   67.996 ms
12  146.188.4.42 (146.188.4.42)  70.045 ms *   71.441 ms
13  * * 146.188.112.82 (146.188.112.82)  69.605 ms
14  * 193.251.129.81 (193.251.129.81)  75.391 ms   74.884 ms
15  193.251.129.41 (193.251.129.41)  93.854 ms   169.094 ms   165.100 ms
16  * * *
17  213.25.5.154 (213.25.5.154)  118.610 ms 213.25.12.154 (213.25.12.154)  122.907 ms
122.927 ms
18  80.50.233.62 (80.50.233.62)  218.868 ms   218.119 ms *
19  80.50.233.62 (80.50.233.62)  123.890 ms   125.167 ms   124.627 ms
```

Enable Mode Commands

This section describes the commands that are available in Enable mode. The Regular mode commands are also available in Enable mode. The Enable Mode Commands table lists the commands that are described in this section.

You must have the enable password to access Enable mode. *See* Switching from Regular Mode to Enable Mode, page 2-4.

Note All of the CLI commands work with both IPv4 and IPv6.

Note All commands and their parameters are case sensitive.

Table 4-2 Enable Mode Commands

Command	Description
access	Manages system access parameters
apache_restart	Restarts the apache server
arp	Displays the ARP table
cache	Manages additional cache operations
config	Enters Configuration mode
current_cli_users	Displays the list of VDS TC admin users who are currently logged into the VDS TC CLI
detection_rules	Manages detection rules configuration
direction	Calculates the visible subnets on the interface
dmesg	Displays the message buffer
eventlog	Provides access to event log operations
exit	Exits the current mode
help	Displays the list of available commands for the mode you are in when you execute the command
ifconfig	Displays the interface(s)
iostat	Displays extended I/O statistics
jumbo	Sends jumbo echo messages
license	Manages the system license
oper	Provides access to system management operations
ping	Sends echo messages
reset	Resets management services
show	Displays run-time information
system	Exports system information
traceroute	Displays the route used by the packet to reach its destination
upgrade	Downloads and installs a software image file
vlan	Adds and removes VLANs from the interface

access

To manage system access parameters, use the **access** command.

access {**enable-password** | **idle-session-timeout** *value* | **user-password**}

Syntax Description

enable-password	Sets the Enable mode password. After you enter the command, the system will prompt you to enter a new password and re-enter the password to confirm it. The new password should be at least four characters long.
idle-session-timeout *value*	Sets the timeout, in seconds, after which the Telnet or SSH session is terminated, both for the Enable mode and the Regular mode commands. The default value is 0 seconds, which disables the timeout feature.
user-password	Sets the Regular mode command user password. After you enter this command, the system will prompt you to first verify the existing password. You will then be prompted to enter a new password and re-enter the password to confirm it. The new password should be at least four characters long The system will verify the password against a set of rules to ensure that the password is complex enough.

Note The default Enable mode password is set during system installation and defaults to the system-id. You can view the system-id using the **show systemid** command in Regular mode. It is strongly recommended that you change the default Enable mode password immediately after the initial installation.

Command Modes

Enable mode

Examples

The following is an example of the dialog that occurs when using the **access enable-password** command:

```
console# access enable-password
New password:
Re-enter new password:
```

The following example sets the idle session timeout value to 32768 seconds:

```
console# access idle-session-timeout 32768
```

The following is an example of the dialog that occurs when using the **access user-password** command:

```
console# access user-password
Changing password for admin.
Old Password:
New Password:
Bad password: too simple
New Password:
Reenter New Password:
Password changed.
```

apache_restart

To restart the apache server, use the **apache_restart** command.

apache_restart

Syntax Description

This command has no arguments or keywords.

Command Modes

Enable mode

Examples

The following is sample output from the **ifconfig** command:

```
console# apache_restart
Restarting httpd2 (SIGHUP)               done
```

arp

To display the ARP table use the **arp** command.

arp

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **arp** command:

```
console> arp
Address     HWtype        HWaddress          Flags Mask  Iface
192.168.0.2 ether         00:17:65:C7:10:42   C          eth0
```

cache

To manage the cache parameters, use the **cache** command..

cache {**active_sessions** *IP_address* | **black_list** {**add** *hash_id* | **dump** | **export** *tftp_server filename* | **remove** *hash_id*} | **hash** *hash_id* | **list** {**display** | **export** *tftp_server filename* | **short**} | **remove hash** *hash_id* | **summary** | **sync** | **volume** {**activate** | **deactivate** | **remove_content**}}

Syntax Description

active_sessions *IP_address*	Displays the active sessions for the IP address referenced with the *IP_address* parameter.

black_list {**add** *hash_id* | **dump** | **export** *tftp_server filename* | **remove** *hash_id*}	• **black_list add** *hash_id:* Adds a file to the black list that has a hash ID that matches the *hash_id* parameter. • **black_list dump**: Displays (dumps) the entire black list. • **black_list export** *tftp_server filename*: Exports the black list to a TFTP server, where *tftp_server* is the hostname or IP address of the TFTP server and *filename* is the name of the to export to. **Note:** The file to which the content is exported must already exist, and must have write access to all. • **remove** *hash_id*: Removes a file from the black list that has a hash ID that matches the *hash_id* parameter.
hash *hash_id*	Dumps the files metadata that has a hash id that matches the *hash_id* parameter.
list {**display** | **export** *tftp_server filename* | **short**}	• **list display**: Displays a full list of cache content. • **list export** *tftp_server filename*: Exports the cache content to a TFTP server, where *tftp_server* is the hostname or IP address of the TFTP server and *filename* is the name of the to export to. • **list short**: Displays the top 1000 Least Recently Used (LRU) cached hash IDs. This process can take about 2 minutes. To interrupt the process, press **Ctrl-C**.
remove hash *hash_id*	Removes a file from cache whose hash ID value matches the *hash_id* parameter. The *hash_id* parameter should match a hash ID that exists in the system cache. For a list of hash IDs stored in the system, use the cache list command.
summary	Displays the CMDB statistics summary.
sync	Verifies and synchronizes the cache metadata. The platform is fully accessible during this process. Note that synching the cache can take a few hours.
volume {**activate** | **deactivate** | **remove_content**}	The **cache volume** command manipulates the cache file system volumes. The **cache volume** commands are mainly used for maintenance purposes, usually for hard drive maintenance. You can view the volumes that can be used for these commands using the **show volumes** command. • **volume activate**: Activates a cache volume. When you enter this command, you will be prompted to enter which volume number you want to activate. • **volume deactivate**: Deactivates a cache volume. When you enter this command, you will be prompted to enter which volume number you want to deactivate. • **volume remove_content**: Removes content from a volume. When you enter this command, you will be prompted to enter the volume number from which you want to remove content.

Command Modes

Enable mode

Examples

The following are examples of managing the black list:

- Adds a file to the black list, based on the hash ID:

```
console# cache black_list add AE7E21FB0CA2DD7464A562E74064248E9B790057
The specified hash was inserted in a black list
console#
console# cache black_list add 6827AC55B43B1B0BAB58FC9F9E7D6B05EF71FDCD
The specified hash was inserted in a black list
```

- Displays the contents of the black list:

```
console# cache black_list dump
HASH                                     PROTOCOL          SIZE            AGE
6827AC55B43B1B0BAB58FC9F9E7D6B05EF71FDCD P2P_SIGNATURE_NA    0               0
AE7E21FB0CA2DD7464A562E74064248E9B790057 P2P_SIGNATURE_NA    0               0
```

- Exports the black list to a TFTP server with in IP address of 192.168.14.26 and a filename of black-list:

```
console# cache black_list export 192.168.14.26 black-list
```

- Removes a file from the black list based on the hash ID:

```
The specified hash was deleted from a black list
console#
console# cache black_list remove 6827AC55B43B1B0BAB58FC9F9E7D6B05EF71FDCD
The specified hash was deleted from a black list
```

The following example displays the metadata for a file based on its hash ID:

```
console# cache hash BCBBAF652BFEAAE3E11C3F279608A1FB7A337DCD
This operation might take some time.(^C to interrupt)
.
.
.
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
HTTP 126B32237A1EDF3508B35B49642945F2E97FB8E2 ST:WATCHED HITS:6 IPS:1 VL:14 SZ:6299870
MP:0 FF:yes FS:1252582601 LS:125267625
1 CO:6 MB FF:100.00 PFF:0.00 checksum 91B6735FB6557E3C959B278C8C0AC2323DD84CC8
Hits: 6
First seen: Thu Sep 10 11:36:41 2009
Last access: Fri Sep 11 13:37:31 2009
Volume id:  14
Cached File size:  6299870
Max piece:  0
Full File:  yes
Full File Size:  6299870
BT_last_start: 0
BT_last_end:   6299870
max known bt piece size 0
cache out in MB 6
File fill factor 100.00
total full pieces 0 (0.00)
HTTP checksum is 91B6735FB6557E3C959B278C8C0AC2323DD84CC8

piece bit mask 0x2
RANGES -------------- RANGES
```

The following are examples of working with the cache content options:

- Displays the cache content:

```
console# cache list display
```

```
This operation might take some time as the whole storage-index is scanned.(^C to
interrupt)
HASH                                     PROTOCOL   SIZE            AGE
D35D8E3D8EEE3BF82D7B8F783FD4D2380A216C67 HTTP       6276762         42
208FD9F7E7C42473291C503931EF22269657285A HTTP       6286497         30
5A88AC47F6B636E5FF6C462D9DE15691A53B26C3 HTTP       6297586         29
26D26351170461CDD3B5680ED0417254C4FA43C1 HTTP       6291394         26
86BA4AC4E0E818C18BD2B9040D7A5E3F19E70776 HTTP       6276664         43
9ADB98FC073B05866EFFBC1159522E9E42261669 HTTP       6295830         43
.
```

- Exports the full list of hash IDs to a TFTP server with an IP address of 192.168.5.117:

```
console# cache list export 192.168.5.117 cache-list
```

- Displays the top 1000 least recently used hash IDs. This operation can take about 2 minutes to complete. Press **CTRL-C** to interrupt the process.

```
console# cache list short
This operation might take some time as the whole storage-index is scanned.(^C to
interrupt)
HASH                                     PROTOCOL   SIZE            AGE
D35D8E3D8EEE3BF82D7B8F783FD4D2380A216C67 HTTP       6276762         42
208FD9F7E7C42473291C503931EF22269657285A HTTP       6286497         30
5A88AC47F6B636E5FF6C462D9DE15691A53B26C3 HTTP       6297586         29
26D26351170461CDD3B5680ED0417254C4FA43C1 HTTP       6291394         26
86BA4AC4E0E818C18BD2B9040D7A5E3F19E70776 HTTP       6276664         43
9ADB98FC073B05866EFFBC1159522E9E42261669 HTTP       6295830         43
94611C230B77D73205BC3090FA2A6104BAEE7990 HTTP       6297222         31
<output omitted>
```

The following example removes a hash ID that is stored in the cache:

```
console# cache remove hash F753B1C31107981BC86D87CF5F7B9EEFD5F5A28B
The specified hash will be deleted in a few minutes
```

The following example starts a cache verification and synchronization process:

```
console# cache sync
Proceeding cache to metadata sync. Some data in the cache might be lost.
Are you sure (y/n)? y
Starting cache synchronization...
```

The following examples show working with volumes:

- Activates a specific volume:

```
console# cache volume activate
Licensed volumes: 120
Please enter volume number <1-120> 4
```

- Deactivates a specific volume:

```
console# cache volume deactivate
Licensed volumes: 120
Please enter volume number <1-120> 4
```

- Removes a specific volume from the CMDB:

```
console# cache volume remove 4
Are you sure? This will remove all hashes from volume 4.
[yes|no] no
```

```
Removing volume 4 has been cancelled
```

config

From Configuration mode you can make platform configuration changes. To enter Configuration mode, use the **config** command.

config

Note For a list of commands available in Configuration mode, *see* Configuration Mode Commands.

Syntax Description This command has no arguments or keywords.

Command Modes Enable mode

Usage Guidelines Changes that you make in Configuration mode are stored but are only implemented when you enter the **apply** command. Use the **exit** command to exit Configuration mode and return to Enable mode.

Examples The following example enters Configurational mode:

```
console# config
configuration#
```

current_cli_users

To display the admin users who are currently logged on, use the **current_cli_users** command.

current_cli_users

Note You must be logged on using an admin username and password to view the output from this command. This command only displays admin users that are logged directly into the VDS TC CLI. It does not display users that have used the sudo command from the Linux CLI to access the VDS TC CLI.

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode and Enable mode

Examples The following example displays a list of users currently logged on to the system:

```
console> current_cli_users
admin    pts/1        Apr  4 21:32 (10.21.150.101)
```

detection_rules

To view detection rules deployment dates and versions and to export the detection rules configurations, use the **detection_rules** command.

detection_rules {**export_groups** *tftp_server filename* | **export_signatures** *tftp_server filename*| **show**}

Syntax Description

export_groups *tftp_server filename*	Exports the group configuration to a TFTP server, where the TFTP server IP address or hostname and filename are specified with the *tftp_server* and *filename* parameters.
export_signatures *tftp_server filename*	Exports the signature configuration to a TFTP server, where the TFTP server IP address or hostname and filename are specified with the *tftp_server* and *filename* parameters.
show	Displays the detection rules versions and deployment dates.

Command Modes

Enable mode

direction

To calculate the visible subnets on the specified interface, use the **direction** command.

direction *interface_name*

Syntax Description

interface_name	The interface for which you want to display the subnets, for example eth0.

Command Modes

Regular mode and Enable mode

Examples

The following sample displays the visible subnets on the interface eth0 using the **direction** command:

```
console> direction eth0
tcpdump: WARNING: eth0: no IPv4 address assigned
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
1000 packets captured
1006 packets received by filter
0 packets dropped by kernel
    453 10.11.12.0/24
    479 10.11.18.0/24
      6 10.11.80.0/24
     59 10.56.10.0/24
      1 64.101.47.0/24
      2 8.8.8.0/24
console>
```

dmesg

To displays the message buffer of the kernel, use the **dmesg** command.

dmesg

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **dmesg** command:

```
console> dmesg
Ending clean XFS mount for filesystem: sdl2
XFS mounting filesystem sdm3
Ending clean XFS mount for filesystem: sdm3
XFS mounting filesystem sdm2
Ending clean XFS mount for filesystem: sdm2
XFS mounting filesystem sdk3
Ending clean XFS mount for filesystem: sdk3
XFS mounting filesystem sdk2
Ending clean XFS mount for filesystem: sdk2
XFS mounting filesystem sdo3
Ending clean XFS mount for filesystem: sdo3
FS mounting filesystem sdo2
Ending clean XFS mount for filesystem: sdo2
XFS mounting filesystem sdn3
Ending clean XFS mount for filesystem: sdn3
XFS mounting filesystem sdn2
Ending clean XFS mount for filesystem: sdn2
monitor[19319]: segfault at 1500 ip 00007fcc02938d9b sp 00007fcbff400040 error 4 in
ld-2.9.so[7fcc02930000+1e000]
device eth0 entered promiscuous mode
device eth0 left promiscuous mode
device eth0 entered promiscuous mode
device eth0 left promiscuous mode
```

eventlog

The event log lists all of the log messages sent to the system log by all of the VDS TC service components (applications, CLI and SNMP). To view or export the content of the event log, use the **eventlog** command. The eventlog command that is available in Enable mode includes additional parameters that are not available in Regular mode. The options are used to duplicate the event log information to an external syslog server.

eventlog {**date** *date*| **export** *tftp_server filename* | **forward** | **show** | **stop** | **tail**}

Syntax Description

date *date*	Displays the eventlog for the date specified with the *date* parameter. The *date* parameter is in the format of DD-MM-YYYY. Press **q** to exit the eventlog and return to the console# prompt.
export *tftp_server filename*	Exports the content of the event log to the TFTP server with the IP address or hostname specified with the *tftp_server* parameter.
forward	Starts eventlog forwarding to a previously configured syslog server.
show	Displays the event log entries. Press **q** to exit the eventlog and return to the console# prompt.
stop	Stops eventlog forwarding to a previously configured syslog server.
tail	Displays the online event log entries.

Command Modes

Regular mode and Enable mode. The **start** and **stop** options are available only in Enable mode.

Usage Guidelines

To add an external syslog server for VDS TC to use, add the following statements to the system configuration:

```
<mgmt-config>
    <external_syslog_ip>ip_address</external_syslog_ip>
</mgmt-config>
```

The *ip_address* parameter specifies the IP address of the external syslog server to use.

Examples

The following example starts forwarding event log messages to the syslog server that is configured in the configuration file:

```
console# eventlog forward
```

The following example displays the eventlog for March 4, 2014:

```
console# eventlog date 4-3-2014
Mar  4 23:33:12 ce-1 pang[12399]: /mnt/vol6        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:33:12 ce-1 pang[12399]: /mnt/vol7        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:33:12 ce-1 pang[12399]: /mnt/vol8        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:33:12 ce-1 pang[12399]: /mnt/vol9        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:33:12 ce-1 pang[12399]: /mnt/vol10       mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:40:19 ce-1 pang[12399]: volume             state      availability owner
total      free       used       usage      total_writes
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol1        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol2        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol3        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol4        mounted          active   ce-1
910        910        0          0.00       0
```

```
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol5        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol6        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol7        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol8        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol9        mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:40:19 ce-1 pang[12399]: /mnt/vol10       mounted          active   ce-1
910        910        0          0.00       0
Mar  4 23:47:25 ce-1 pang[12399]: volume             state      availability owner
total      free       used       usage      total_writes
```

The following example stops forwarding event log messages to the configured syslog server:

```
console# eventlog stop
```

exit

To exit any mode or close an active CLI session use the **exit** command. In Enable mode, this command returns the user to Regular mode. In Regular mode, this command terminates the session and the user is logged out of the CLI session.

exit

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode, Enable mode, and Configuration mode

Examples

The following example shows how to exit the current session:

```
console> exit
```

help

To display the CLI commands that are available in the current mode, with a short description of each command, use the **help** command. For example, if you enter the **help** command in Enable mode, then the command displays only the commands that are available in Enable mode with a short description of each command.

help

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode, Enable mode, and Configuration mode

Examples

The following example displays the list of CLI commands that are available in Regular mode:

```
console# ?
access               Manage system access parameters
apache_restart       Restart apache
arp                  Show arp table
cache                Cache operations
config               Enter configuration mode
current_cli_users   Show currently logged in cli users
detection_rules      Manage detection rules configuration
direction            Calculate seen subnets on interface (IPv4 traffic only)
dmesg                Display dmesg
eventlog             Event log commands
exit                 Exit current mode
help                 Commands description
ifconfig             Display interface(s)
iostat               Display IO statistics
jumbo                Send jumbo echo messages
license              Manage system license
oper                 System management operations
ping                 Send echo messages
reset                Reset management service
show                 Show run-time information
system               System information
traceroute           Print the route packets take to network host
upgrade              Download and install software image file
vlan                 VLAN operations
```

ifconfig

To display the details of the interfaces, use the **ifconfig** command.

ifconfig

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **ifconfig** command:

```
console# ifconfig
bond0     Link encap:Ethernet  HWaddr D4:8C:B5:4D:C0:14
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:37161009 errors:0 dropped:0 overruns:0 frame:0
          TX packets:953496 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:3045286713 (2904.2 Mb)  TX bytes:245299515 (233.9 Mb)

bond0:1   Link encap:Ethernet  HWaddr D4:8C:B5:4D:C0:14
          inet addr:10.56.194.36  Bcast:10.56.195.255  Mask:255.255.254.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1

eth0      Link encap:Ethernet  HWaddr D4:8C:B5:4D:C0:14
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:19175744 errors:0 dropped:0 overruns:0 frame:0
```

```
          TX packets:953496 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1907814608 (1819.4 Mb)  TX bytes:245299515 (233.9 Mb)
eth1      Link encap:Ethernet  HWaddr D4:8C:B5:4D:C0:14
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:17985265 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1137472105 (1084.7 Mb)  TX bytes:0 (0.0 b)

        <output omitted>
```

iostat

To report CPU statistics and input/output statistics for devices and partitions, use the **iostat** command.

iostat [**-t** *interval* [**-k** *count*]]

Syntax Description

-t *interval*	The amount of time, in seconds between each report. The default is 5 seconds.
-k *count*	Used in conjunction with the interval parameter. If the count parameter is specified, the count determines the number of reports generated at the specified interval. If the interval parameter is specified without the count parameter, the **iostat** command generates reports continuously until you press **Ctrl-C**.

Command Modes

Regular mode and Enable mode

Examples

The following example generates two I/O statistics reports four seconds apart:

```
console# iostat -t 2 -k 4
Linux 2.6.21-affined-8-default (mg-1)   04/22/09

Time: 13:18:42
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          11.29    0.03    6.02    1.38    0.00   81.28

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.50 120.18  0.62  9.44   98.11 1050.22    49.06   525.11   114.11     0.68
67.44   5.78   5.82

Time: 13:18:46
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           9.48    0.00    6.11    1.68    0.00   82.73

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.00  41.79  0.00 10.95    0.00  429.85     0.00   214.93    39.27     0.13
11.82   6.36   6.97

Time: 13:18:50
```

```
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          10.38    0.00    6.25    1.25    0.00   82.12

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.00  27.50  0.00  8.75    0.00  298.00     0.00   149.00    34.06     0.10
11.66   6.51   5.70

Time: 13:18:54
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           9.05    0.00    6.05    1.00    0.00   83.90

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.00  30.67  0.00  7.98    0.00  313.22     0.00   156.61    39.25     0.10
12.00   5.25   4.19
```

jumbo

To send jumbo echo messages, use the **iostat** command.

jumbo [**-c** *counter*] [**-I** {*IP_address* | *interface*}] *destination*

Syntax Description

-c *counter*	The number of times the request is generated.
-I {*IP_address* | *interface*}	The interface IP address or interface name from which the echo requests are sent.
destination	The destination to which the jumbo echo message will be sent.

Command Modes

Regular mode and Enable mode

Examples

The following example sends jumbo echo messages to 192.168.5.117 sourced from the eth0 interface:

```
console> jumbo -I eth0 192.168.5.117
PING 192.168.5.117 (192.168.5.117) from 192.168.5.117 eth0: 8972(9000) bytes of data.
8980 bytes from 192.168.5.117: icmp_seq=1 ttl=64 time=0.043 ms
8980 bytes from 192.168.5.117: icmp_seq=2 ttl=64 time=0.024 ms
8980 bytes from 192.168.5.117: icmp_seq=3 ttl=64 time=0.028 ms
8980 bytes from 192.168.5.117: icmp_seq=4 ttl=64 time=0.033 ms
8980 bytes from 192.168.5.117: icmp_seq=5 ttl=64 time=0.039 ms

--- 192.168.5.117 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3998ms
rtt min/avg/max/mdev = 0.024/0.033/0.043/0.008 ms
```

The following example sends two jumbo echo messages to 192.168.3.170 sourced from the eth0 interface:

```
console> jumbo -c 2 -I eth0 192.168.3.170
PING 192.168.3.170 (192.168.3.170) from 192.168.5.117 eth0: 8972(9000) bytes of data.
From 192.168.5.117 icmp_seq=1 Frag needed and DF set (mtu = 1500)
From 192.168.5.117 icmp_seq=1 Frag needed and DF set (mtu = 1500)
```

```
--- 192.168.3.170 ping statistics ---
0 packets transmitted, 0 received, +2 errors
```

license

To manage VDS-TC system license, use the **license** command. This license controls operational parameters, such as the supported protocols and features, and the maximum cache bandwidth.

license {**activate** | **import** *tftp_server filename* | **show**}

Syntax Description

activate	Activates the system license.
import *tftp_server filename*	Imports a license from the TFTP server where the server name or ip address and file location are specified with the *tftp_server* and *filename* parameters.
show	Displays the currently licensed operational parameters.

Command Modes

Enable mode

Examples

The following example activates the installed license:

```
console# license activate
Licensed chassis serial number: DGB9C4J
EDK enabled:                    1
Bittorent enabled:              1
Kazaa enabled:                  1
Gnutella enabled:               1
Ares enabled:                   1
Http enabled:                   1
Thunder enabled:                0
Storage volumes:                12
Controllers:                    0
Evaluation ends on:              3-10-2014
Max bandwidth:                  1000 Mbps
```

The following example imports the license from the TFTP server with an IP address of 10.1.1.65:

```
console# license import 10.1.1.65 SF-PALicense.xml
Licensed chassis serial number: DGB9C4J
EDK enabled:                    1
Bittorent enabled:              1
Kazaa enabled:                  1
Gnutella enabled:               1
Ares enabled:                   1
Http enabled:                   1
Thunder enabled:                0
Storage volumes:                12
Controllers:                    0
Evaluaton ends on:              3-10-2014
Max bandwidth:                  1000 Mbps
```

The following example displays the installed license operational parameters:

```
console# license show
Licensed chassis serial number: DGB9C4J
EDK enabled:                    1
Bittorent enabled:              1
Kazaa enabled:                  1
Gnutella enabled:               1
Ares enabled:                   1
Http enabled:                   1
Thunder enabled:                1
Storage volumes:                12
Controllers:                    0
Evaluaton ends on:              3-10-2014
Max bandwidth:                  1000 Mbps
```

oper service

To control the running state of the platform, including starting, stopping, or restarting the platform software and all its services, use the **oper service** command.

oper service {**powerdown** | **powerup** | **start** | **stop**}

Syntax Description

powerdown	Service power down.
powerup	Service power up.
start	Starts the VDS TC software and services.
stop	Stops the VDS TC software and services

Command Modes

Enable mode

Examples

```
console# oper service stop
Are you sure (y/n)? y
Stopping service
console# exit
console> show status
Cluster state: disabled

Server Slot   Status   Operational state   Device state   Administrative state
ce-1         powered on       enabled       started              unlocked
```

The following example stars the VDS TC software and its services:

```
console# oper service start
Starting service
console# show status
Operational state    Device state         Administrative state
 enabled              started              unlocked
```

ping

To diagnose basic network connectivity, use the **ping** command. The **ping** command uses the ICMP protocol mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway, and displays the round-trip time for the echo response to arrive back to the server on which the command was issued.

Note To abort the ping command, press **Ctrl-C**.

ping [**-c** *counter*] [**-I** {*IP_address* | *interface*}] *destination*

Syntax Description

-c *counter*	The number of ICMP echo requests to be sent to the destination address. If you do not specify the number of echo requests to send, the ping will continue until you press **Ctrl-C**.
-I {*IP_address* | *interface*}	The interface IP address or interface name from which the pings are sent.
destination	The destination to which the ping messages will be sent. This value can be an IP address or a hostname.

Command Modes

Regular mode and Enable mode

Usage Guidelines

To abort the ping command, press **Ctrl-C**.

Examples

The following example pings the destination at 192.168.0.202 sourced from the eth0 interface until Ctrl-C is pressed:

```
console> ping -I eth0 192.168.0.202
PING 192.168.0.202 (192.168.0.202) from 192.168.0.202 eth0: 56(84) bytes of data.
64 bytes from 192.168.0.202: icmp_seq=1 ttl=64 time=0.023 ms
64 bytes from 192.168.0.202: icmp_seq=2 ttl=64 time=0.035 ms
64 bytes from 192.168.0.202: icmp_seq=3 ttl=64 time=0.028 ms
64 bytes from 192.168.0.202: icmp_seq=4 ttl=64 time=0.020 ms
64 bytes from 192.168.0.202: icmp_seq=5 ttl=64 time=0.029 ms
64 bytes from 192.168.0.202: icmp_seq=6 ttl=64 time=0.030 ms
64 bytes from 192.168.0.202: icmp_seq=7 ttl=64 time=0.032 ms
64 bytes from 192.168.0.202: icmp_seq=8 ttl=64 time=0.029 ms
64 bytes from 192.168.0.202: icmp_seq=9 ttl=64 time=0.030 ms
64 bytes from 192.168.0.202: icmp_seq=10 ttl=64 time=0.030 ms
--- 192.168.0.202 ping statistics ---
10 packets transmitted, 0 received, 0% packet loss, time 19023ms
rtt min/avg/max/mdev = 0.019/0.027/0.037/0.006 ms
```

reset

To reset the management services, use the **reset** command.

Note Resetting the management services will disconnect *your* current administration session, and you will have to login again.

reset

Syntax Description This command has no arguments or keywords.

Command Modes Enable mode

Examples The following example resets the management service and all of its services:

```
console# reset
Are you sure (y/n)? y
.
.Connection terminated
```

show

To display run-time information related to the operational environment of VDS TC, use the **show** command. The Enable mode show command includes all of the parameters available in Regular mode (*see* show, page 4-11) and the following additional parameters:

show {**config** | **license**}

Syntax Description

config	Displays the running configuration.
license	Displays the system license information.

Note Additional parameters for this command are available in Regular mode. Refer to the **show** command in the Regular Mode Commands section for a description of these parameters.

Command Modes Enable mode

Examples The following is sample output from the **show config** command:

```
console# show config
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>

<cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='./cluster_conf.xsd'>
  <mgmt-config>
```

```
  <ipaddr>10.56.194.36</ipaddr>
  <netmask>255.255.254.0</netmask>
  <default-gw>10.56.194.1</default-gw>
  <nameserver>8.8.8.8</nameserver>
  <site_name>CISCO UCS240 UB1K</site_name>
</mgmt-config>
<web-config></web-config>
<common>
  <snmp>
    <trap-ip>10.11.12.1</trap-ip>
    <snmp-read-community>gdcbhv</snmp-read-community>
    <snmp-write-community>nkppui</snmp-write-community>
    <snmp-trap-community>ffff</snmp-trap-community>
  </snmp>
  <service>
    <protocols>
      <enable-bittorent>1</enable-bittorent>
      <enable-edk>1</enable-edk>
      <enable-http>1</enable-http>
      <enable-ares>1</enable-ares>
      <enable_cache_out_port>1</enable_cache_out_port>
    </protocols>
    <net>
      <fwd-mode>BOUNCING</fwd-mode>
      <bounce id='0'></bounce>
      <subnet_range_per_link name='a'>
        <cidr_range>10.10.0.0/16</cidr_range>
      </subnet_range_per_link>
    </net>
  </service>
</common>
<blades>1</blades>
<blade id='1'>
  <cache-engine>
    <network>
      <network_interfaces number='8'>
        <nic nic_index='0'>
          <name>eth4</name>
          <nic_detail>IFF_PF_PACKET</nic_detail>
          <vip>10.138.201.1</vip>
        </nic>
        <nic nic_index='1'>
          <name>eth5</name>
          <nic_detail>IFF_PF_PACKET</nic_detail>
          <vip>10.138.201.2</vip>
        </nic>
        <nic nic_index='2'>
          <name>eth6</name>
          <nic_detail>IFF_PF_PACKET</nic_detail>
          <vip>10.138.201.3</vip>
        </nic>
        <nic nic_index='3'>
          <name>eth7</name>
          <nic_detail>IFF_PF_PACKET</nic_detail>
          <vip>10.138.201.4</vip>
        </nic>
        <nic nic_index='4'>
          <name>eth2</name>
          <nic_detail>IFF_PF_PACKET</nic_detail>
        </nic>
        <nic nic_index='5'>
          <name>eth2</name>
          <nic_detail>IFF_PF_PACKET</nic_detail>
        </nic>
```

```
            <nic nic_index='6'>
              <name>eth3</name>
              <nic_detail>IFF_PF_PACKET</nic_detail>
            </nic>
            <nic nic_index='7'>
              <name>eth3</name>
              <nic_detail>IFF_PF_PACKET</nic_detail>
            </nic>
          </network_interfaces>
        </network>
        <service>
          <net>
            <cacheout_bypass_details name='1-first' nic_index='4'>
              <bypass_netmask>255.225.255.0</bypass_netmask>
              <bypass_remote_server_ip1>10.138.31.254</bypass_remote_server_ip1>
              <associated_network_element_index>0</associated_network_element_index>
              <bypass_local_ip>10.138.31.1</bypass_local_ip>
            </cacheout_bypass_details>
            <cacheout_bypass_details name='1-second' nic_index='5'>
              <bypass_netmask>255.225.255.0</bypass_netmask>
              <bypass_remote_server_ip1>10.138.31.254</bypass_remote_server_ip1>
              <associated_network_element_index>1</associated_network_element_index>
              <bypass_local_ip>10.138.31.2</bypass_local_ip>
            </cacheout_bypass_details>
            <cacheout_bypass_details name='2-first' nic_index='6'>
              <bypass_netmask>255.225.255.0</bypass_netmask>
              <bypass_remote_server_ip1>10.138.31.254</bypass_remote_server_ip1>
              <associated_network_element_index>2</associated_network_element_index>
              <bypass_local_ip>10.138.31.3</bypass_local_ip>
            </cacheout_bypass_details>
            <cacheout_bypass_details name='2-second' nic_index='7'>
              <bypass_netmask>255.225.255.0</bypass_netmask>
              <bypass_remote_server_ip1>10.138.31.254</bypass_remote_server_ip1>
              <associated_network_element_index>3</associated_network_element_index>
              <bypass_local_ip>10.138.31.4</bypass_local_ip>
            </cacheout_bypass_details>
          </net>
        </service>
      </cache-engine>
    </blade>
  </cluster>
```


The following is example output from the **show license** command:

```
console# show license
Licensed chassis serial number: DGB9C4J
EDK enabled:                    1
Bittorent enabled:              1
Kazaa enabled:                  1
Gnutella enabled:               1
Ares enabled:                   1
Http enabled:                   1
Thunder enabled:                0
Storage volumes:                12
Controllers:                    0
Evaluaton ends on:              3-10-2014
Max bandwidth:                  1000 Mbps
```

system

To export system log or system statistics information, use the **system** command. This command contains the following parameters:

system {**logs export** | **statistics export**}

Syntax Description

logs export	Exports the system logs to a TFTP server
statistics export	Exports the system statistics to a TFTP server

Command Modes

Enable mode

traceroute

To discover the IP routes that packets will actually take when traveling to their destination, use the **traceroute** command. The **traceroute** command tracks the route of a packet across a TCP/IP network on its way to a given host. It utilizes the IP protocol time to live (TTL) field and attempts to elicit an ICMP TIME_EXCEEDED response from each gateway along the path to the destination.

traceroute [**-n**] *destination*

Syntax Description

-n	Forces the traceroute command to avoid mapping IP addresses to host names when displaying the output.
destination	The name or IP address of the destination.

Command Modes

Regular mode and Enable mode

Examples

The following example shows a traceroute to the destination 213.25.17.65:

```
console# traceroute 213.25.17.65
traceroute to 213.25.17.65 (213.25.17.65), 30 hops max, 40 byte packets
 1  192.168.0.2 (192.168.0.2)  1.316 ms   1.485 ms   1.292 ms
 2  10.1.2.253 (10.1.2.253)  0.656 ms   0.372 ms   0.747 ms
 3  212.150.66.65 (212.150.66.65)  1.481 ms   1.210 ms   1.044 ms
 4  212.150.6.137 (212.150.6.137)  5.517 ms   6.398 ms   6.574 ms
 5  gig0-1-gw1.thc.nv.net (212.143.200.33)  10.862 ms   10.842 ms   10.668 ms
 6  * coresw1-hfa-gw1.thc.nv.net (212.143.200.1)  6.721 ms   6.895 ms
 7  teng2-1-0.gw2.hfa.nv.net (212.143.8.196)  5.959 ms   9.117 ms   8.460 ms
 8  pos0-1-1.brdr2.lnd.nv.net (212.143.12.57)  77.423 ms *   75.177 ms
 9  GigabitEthernet3-1.GW1.LND9.ALTER.NET (146.188.55.61)  73.425 ms   69.143 ms 70.365 ms
10  * so-3-0-0.CR2.LND9.ALTER.NET (158.43.150.145)  157.824 ms   153.850 ms
11  so-0-3-0.XT2.LND2.ALTER.NET (146.188.15.245)  69.099 ms   69.834 ms *
12  GigabitEthernet1-0-0.BR1.LND18.ALTER.NET (146.188.4.42)  73.002 ms   72.350 ms
78.342 ms
```

```
13  GE.LND.opentransit.net (146.188.112.82)  70.164 ms   69.356 ms   69.688 ms
14  tengige0-8-0-0.lontr1.London.opentransit.net (193.251.129.81)  80.167 ms   80.134 ms
79.107 ms
15  xe-0-0-1-0.ffttr2.Frankfurt.opentransit.net (193.251.129.41)  92.546 ms   95.034 ms
94.282 ms
16  * * *
17  do.wro-ar3.z.wro-r1.tpnet.pl (213.25.5.154)  120.296 ms do.wro-ar3.z.wro-r2.tpnet.pl
(213.25.12.154)  134.187 ms   130.254 ms
18  * z-easy-com.wro-ar1.tpnet.pl (80.50.233.62)  236.345 ms   235.563 ms
19  z-easy-com.wro-ar1.tpnet.pl (80.50.233.62)  124.651 ms   125.124 ms   126.872 ms
```

upgrade

To upgrade the software version of VDS TC, use the **upgrade** command.

upgrade *tftp_server file*

Syntax Description

`tftp_server`	The hostname or IP address of the TFTP server. This server must be accessible from the VDS TC platform on which you are running the **upgrade** command.
`file`	The name of the file containing the software version package received from Cisco. **Note:** If the TFTP server is running on one of the Cisco servers, the upgrade command attempts to retrieve the file from the /tftpboot folder.

Command Modes

Enable mode

Note For complete details on upgrading your VDS TC software, see the *Cisco Videoscape Distribution Suite Transparent Caching Application Upgrade Guide* at http://www.cisco.com/c/dam/en/us/td/docs/video/videoscape/distribution_suite/vds/v5_7_3/VDS-TC_5.7.3_app_upgrade_guide.pdf.

vlan

To add or remove a VLAN to or from an interface, use the **vlan** command

vlan {**add** *interface_name vlan_id ip* [*mask*] | **remove** *interface_name vlan_id*}

Syntax Description

`add` *`interface_name vlan_id ip [mask]`*	Adds the new vlan specified with the *vlan_id* parameter to the interface indicated with the *interface_name* parameter. You must specify an IP address for the interface.
`remove` *`interface_name vlan_id`*	Removes the vlan specified with the *vlan_id* parameter from the interface indicated with the *interface_name* parameter.

Command Modes Enable mode

Examples The following example adds vlan 10 to eth0 with an IP address 10.11.12.15 and subnet mask of 255.255.255.0:

```
console# vlan add eth0 10 10.11.12.15 255.255.255.0
Set name-type for VLAN subsystem. Should be visible in /proc/net/vlan/config
Added VLAN with VID == 10 to IF -:eth0:-
interface eth0.10 is up
```

The following example removes vlan 10 from eth0:

```
console# vlan remove eth0 10
Removed VLAN -:eth0.10:-
eth0.10 removed
```

Configuration Mode Commands

This section describes the commands that are available in Configuration mode. The Configuration Mode Commands table lists the commands that are described in this section.

Table 4-3 Configuration Mode Commands

Command	Description
apply	Applies the pending configuration changes.
diff	Shows the pending configuration changes.
discard	Discards the pending changes.
display	Displays the current configuration.
exit	Exits Configuration mode.
export	Exports the cluster configuration to the TFTP server.
help	Displays the command syntax for each configuration command.
import	Imports the cluster configuration from the TFTP server.
network	Configures the management network interface.
restore	Restores the last good configuration.
time	Sets the system date and time.

apply

To apply the pending configuration changes, use the **apply** command. This command immediately applies the configuration changes to the live platform.

apply

Syntax Description This command has no arguments or keywords.

Command Modes Configuration mode

Usage Guidelines If there are no configuration changes to apply, this command will return the message “Configurations are identical.”.

Examples The following is sample output from the **apply** command:

```
configuration# apply
applying configuration...
Configuration applied
```

diff

To display the proposed configuration changes, use the **diff** command.

diff

Syntax Description This command has no arguments or keywords.

Command Modes Configuration mode

Usage Guidelines When you enter the **diff** command, the new configuration parameters are indicated by a plus (**+**) sign as the first character on the line, while the current configuration parameters are indicated by a minus (**-**) sign as the first character on the line. If there are no proposed configuration changes to display, the command returns the message “Configurations are identical”.

Examples The following is sample output from the **diff** command:

```
configuration# diff
  <?xml version="1.0" encoding="UTF-8" standalone="no" ?>

<cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='cluster_conf.xsd'>
   <mgmt-config>
      <ipaddr>192.168.0.202</ipaddr>
      <netmask>255.255.255.0</netmask>
      <default-gw>192.168.0.2</default-gw>
      <nameserver>10.1.1.235</nameserver>
      <alert-email>support@cisco.com</alert-email>
      <site_name>>192.168.0.202 </site_name>
      <external_syslog_ip>127.0.0.1</external_syslog_ip>
    </mgmt-config>
    <web-config></web-config>
    <common>
      <snmp>
        <trap-ip>aa.bb.cc.dd</trap-ip>
        <snmp-read-community>gdcbhv</snmp-read-community>
        <snmp-write-community>nkppui</snmp-write-community>
```

```
      <snmp-trap-community>nkppui</snmp-trap-community>
    </snmp>
    <service>
      <protocols>
-       <enable-bittorent>1</enable-bittorent>
+       <enable-bittorent>0</enable-bittorent>
        <enable-edk>1</enable-edk>
        <enable-http>1</enable-http>
        <enable-ares>1</enable-ares>

      </protocols>
      <net>
        <fwd-mode>PROMISC</fwd-mode>
        <bridge id='0'>
          <interface-world>iff2</interface-world>
          <interface-isp>iff1</interface-isp>
        </bridge>
      </net>
    </service>
  </common>
  <blades>1</blades>
  <blade id='1'>
    <cache-engine>
      <network></network>
    </cache-engine>
  </blade>
</cluster>
```

discard

To discard any configuration change requested since entering Configuration mode, use the **discard** command.

discard

Syntax Description

This command has no arguments or keywords.

Command Modes

Configuration mode

Examples

The following is sample output from the **discard** command:

```
configuration# discard
```

display

To display the current configuration, use the **display** command.

display

Syntax Description

This command has no arguments or keywords.

Command Modes

Configuration mode

Examples

The following is sample output from the **display** command:

```
configuration# display
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>

<cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='./cluster_conf.xsd'>
  <mgmt-config>
    <ipaddr>10.56.194.36</ipaddr>
    <netmask>255.255.254.0</netmask>
    <default-gw>10.56.194.1</default-gw>
    <nameserver>8.8.8.8</nameserver>
    <site_name>CISCO UCS240 UB1K</site_name>
  </mgmt-config>
  <web-config></web-config>
  <common>
    <snmp>
      <trap-ip>10.11.12.1</trap-ip>
      <snmp-read-community>gdcbhv</snmp-read-community>
      <snmp-write-community>nkppui</snmp-write-community>
      <snmp-trap-community>ffff</snmp-trap-community>
    </snmp>
    <service>
      <protocols>
        <enable-bittorent>1</enable-bittorent>
        <enable-edk>1</enable-edk>
        <enable-http>1</enable-http>
        <enable-ares>1</enable-ares>
        <enable_cache_out_port>1</enable_cache_out_port>
      </protocols>
      <net>
        <fwd-mode>BOUNCING</fwd-mode>
        <bounce id='0'></bounce>
        <subnet_range_per_link name='a'>
          <cidr_range>10.10.0.0/16</cidr_range>
        </subnet_range_per_link>
      </net>
    </service>
  </common>
  <blades>1</blades>
  <blade id='1'>
    <cache-engine>
      <network>
        <network_interfaces number='8'>
          <nic nic_index='0'>
            <name>eth4</name>
            <nic_detail>IFF_PF_PACKET</nic_detail>
            <vip>10.138.201.1</vip>
          </nic>
          <nic nic_index='1'>
            <name>eth5</name>
            <nic_detail>IFF_PF_PACKET</nic_detail>
            <vip>10.138.201.2</vip>
          </nic>
          <nic nic_index='2'>
            <name>eth6</name>
            <nic_detail>IFF_PF_PACKET</nic_detail>
            <vip>10.138.201.3</vip>
          </nic>
```

```
            <nic nic_index='3'>
              <name>eth7</name>
              <nic_detail>IFF_PF_PACKET</nic_detail>
              <vip>10.138.201.4</vip>
            </nic>
            <nic nic_index='4'>
              <name>eth2</name>
              <nic_detail>IFF_PF_PACKET</nic_detail>
            </nic>
            <nic nic_index='5'>
              <name>eth2</name>
              <nic_detail>IFF_PF_PACKET</nic_detail>
            </nic>
            <nic nic_index='6'>
              <name>eth3</name>
              <nic_detail>IFF_PF_PACKET</nic_detail>
            </nic>
            <nic nic_index='7'>
              <name>eth3</name>
              <nic_detail>IFF_PF_PACKET</nic_detail>
            </nic>
          </network_interfaces>
        </network>
        <service>
          <net>
            <cacheout_bypass_details name='1-first' nic_index='4'>
              <bypass_netmask>255.225.255.0</bypass_netmask>
              <bypass_remote_server_ip1>10.138.31.254</bypass_remote_server_ip1>
              <associated_network_element_index>0</associated_network_element_index>
              <bypass_local_ip>10.138.31.1</bypass_local_ip>
            </cacheout_bypass_details>
            <cacheout_bypass_details name='1-second' nic_index='5'>
              <bypass_netmask>255.225.255.0</bypass_netmask>
              <bypass_remote_server_ip1>10.138.31.254</bypass_remote_server_ip1>
              <associated_network_element_index>1</associated_network_element_index>
              <bypass_local_ip>10.138.31.2</bypass_local_ip>
            </cacheout_bypass_details>
            <cacheout_bypass_details name='2-first' nic_index='6'>
              <bypass_netmask>255.225.255.0</bypass_netmask>
              <bypass_remote_server_ip1>10.138.31.254</bypass_remote_server_ip1>
              <associated_network_element_index>2</associated_network_element_index>
              <bypass_local_ip>10.138.31.3</bypass_local_ip>
            </cacheout_bypass_details>
            <cacheout_bypass_details name='2-second' nic_index='7'>
              <bypass_netmask>255.225.255.0</bypass_netmask>
              <bypass_remote_server_ip1>10.138.31.254</bypass_remote_server_ip1>
              <associated_network_element_index>3</associated_network_element_index>
              <bypass_local_ip>10.138.31.4</bypass_local_ip>
            </cacheout_bypass_details>
          </net>
        </service>
      </cache-engine>
    </blade>
  </cluster>
```


exit

To exit any mode or close an active CLI session use the **exit** command. When you execute this command in Configuration mode, you are returned to the Enable mode.

exit

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode, Enable mode, and Configuration mode

Usage Guidelines

When you execute this command in Configuration mode, if you have created any configuration change but did not use the **apply** command to implement the changes, the following warning message appears:

```
Exiting configuration mode without apply, will discard changes.
   Are you sure? [N/y] n
```

Examples

The following example shows how to exit Configuration mode:

```
Configuration# exit
```

export

To export the current configuration to a TFTP server, use the **export** command.

export *tftp_server filename*

Syntax Description

tftp_server	The IP address or the hostname of the TFTP server to export the configuration to.
filename	The filename to create when exporting the current configuration file.

Command Modes

Configuration mode

Examples

The following example exports the current configuration to the TFTP server with an IP address of 192.168.0.97:

```
configuration# export 192.168.0.97 current-config
```

Help

To display the CLI commands that are available in the current mode, with a short description of each command, use the **help** command.

help

Syntax Description

This command has no arguments or keywords.

Command Modes Regular mode, Enable mode, and Configuration mode

Examples The following example displays the list of CLI commands that are available in Configuration mode:

```
configuration# ?
apply                Apply config changes
diff                 Show pending changes
discard              Discard pending changes
display              Display pending configuration
exit                 Exit current mode
export               Export cluster configuration to TFTP server
help                 Commands description
import               Import cluster configuration from TFTP server
network              Configure management network interface
restore              Restore last good configuration
time                 Set system date and time
```

import

To import a configuration from a TFTP server, use the **import** command.

import *tftp_server filename*

Syntax Description

tftp_server	The IP address or the hostname of the TFTP server from which to import the configuration. **Note:** You can use **localhost** for the *tftp_server* parameter. If you use **localhost**, then the file must be located in the /tftpboot folder of the VDS TC platform to which you are importing.
filename	The filename that contains the configuration that you want to import.

Command Modes Configuration mode

Examples The following example imports a new configuration to the VDS TC platform from the TFTP server 192.168.0.97 and uses the **diff** command to show the differences in the imported configuration file:

```
configuration# import 192.168.0.97 current-config
configuration# diff
  <?xml version="1.0" encoding="UTF-8" standalone="no" ?>

  <cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='cluster_conf.xsd'>
    <mgmt-config>
      <ipaddr>192.168.0.97</ipaddr>
      <netmask>255.255.255.0</netmask>
-     <default-gw>192.168.0.1</default-gw>
+     <default-gw>192.168.0.2</default-gw>
      <nameserver>194.90.1.5</nameserver>
      <alert-email>support@cisco.com</alert-email>
      <site_name>UB1K office - promisc </site_name>
```

```
</mgmt-config>
<web-config></web-config>
```

network

To change the configuration default gateway or the configuration management network IP address, use the **network** command.

network {**default6_gw** *dg_ipv6_address* | **default_gw** *dg_ipv4_address* | **ip** *ipv4_address netmask* | **ip6** *ipv6_address/ipv6_prefix*}

Note Changing the ip address or default gateway could be dangerous, because if the change is done from a Telnet or SSH session to a remote VDS-TC server, it might end the connection with which you are currently working.

Syntax Description

default6_gw *dg_ipv6_address*	Sets the default IPv6 gateway of the VDS TC platform to the address specified with the *dg_ip_address* parameter.
default_gw *dg_ip_address*	Sets the default IPv4 gateway of the VDS TC platform to the address specified with the *dg_ip_address* parameter.
ip *ip_address netmask*	Changes the management network interface IP address to the address and subnet mask specified with the provided parameters.
ip6 *ip_v6address/ipv6_prefix*	Configures the management network interface IPv6 address and prefix.

Command Modes

Configuration mode

Examples

The following example sets the default gateway for the VDS TC platform to 192.168.0.2 and sets the management network IP address to 192.168.0.97:

```
configuration# network default_gw 192.168.0.2
configuration# network ip 192.168.0.97 255.255.255.0
```

restore

To restore the previous configuration, use the **restore** command. You must still use the **apply** command to apply the restored configuration to the system.

restore

Syntax Description

This command has no arguments or keywords.

Command Modes Configuration mode

Examples The following example restores the previous configuration. Note that restoring the configuration still requires the use of the **apply** command to implement the previous:

```
configuration# restore
configuration# diff
  <?xml version="1.0" encoding="UTF-8" standalone="no" ?>
  <cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='cluster_conf.xsd'>
    <mgmt-config>
      <ipaddr>192.168.0.202</ipaddr>
      <netmask>255.255.255.0</netmask>
  -    <default-gw>192.168.0.2</default-gw
  +    <default-gw>192.168.0.3</default-gw>
      <nameserver>10.1.1.235</nameserver>
      <alert-email>support@cisco.com</alert-email>
      <site_name>192.168.0.202 </site_name>
      <external_syslog_ip>127.0.0.1</external_syslog_ip>
    </mgmt-config>
    <web-config></web-config>
    <common>
```

time

To change the system time, use the **time** command.

time *MMDDYYhhmm*

Syntax Description

MMDDYYhhmm	Specifies the new date and time to set on the system.

Command Modes Configuration mode

Examples The following example sets the date and time to May 26, 2012 10:27 am:

```
configuration# time 0331142124
Mon Mar 31 21:24:00 GMT-2 2014
```

Monitoring VDS TC

You can view statistical information regarding bandwidth utilization, caching statistics, and server status for VDS-TC using both SNMP and the CLI. This chapter describes the CLI commands that you can use to monitor VDS TC.

show config

To view the current active configuration in XML format, use the **show config** command.

show config

Syntax Description This command has no arguments or keywords.

Command Modes Enable mode

show eth_status

To view the eth status, use the **show eth_status** command.

show eth_status

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode and Enable mode

Examples The following example shows output from the **show eth_status** command.

```
console> show eth_status
Blade      eth1    eth2    eth3    eth4    eth5    eth6    eth7
ce-1       UP      UP      UP      UP      UP      UP      UP
```

show eventlog

To display the contents of the event log, use the **show eventlog** command.

show eventlog

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **show eventlog** command:

```
console> show eventlog
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol9        mounted          active   ce-1
678        546        131        19.37
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol10       mounted          active   ce-1
678        548        129        19.15
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol11       mounted          active   ce-1
678        546        132        19.51
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol12       mounted          active   ce-1
678        546        131        19.44
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol13       mounted          active   ce-1
678        546        132        19.49
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol14       mounted          active   ce-1
678        544        134        19.76
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol15       mounted          active   ce-1
678        543        135        19.94
Apr 22 10:06:59 ce-1 pang[29533]: volume             state      availability owner
total      free       used       usage
<output committed>
```

show license

To view the current active license, use the **show license** command.

show license

Syntax Description

This command has no arguments or keywords.

Command Modes

Enable mode

Examples

The following example shows output from the **show license** command.

```
console# show license
Licensed chassis serial number: DGB9C4J
EDK enabled:                    1
Bittorent enabled:              1
Kazaa enabled:                  1
Gnutella enabled:               1
```

```
Ares enabled:                   1
Http enabled:                   1
Thunder enabled:                0
Storage volumes:                15
Controllers:                    0
Evaluaton ends on:              3-10-2009
Max bandwidth:                  1000 Mbps
```

show status

To view the service status, physical slot status, administration status (enabled/disabled) and operational status, use the **show status** command.

show status

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **show status** command:

```
Console> show status
Operational state    Device state         Administrative state
 enabled              started              unlocked
```

show systemid

To view the unique system identifier, which is used for support and licensing purposes, use the **show systemid** command.

show systemid

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **show systemid** command:

```
console> show systemid
*8GB9C4J
```

show time

To view the system date and time, use the **show time** command.

show time

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode and Enable mode

Examples The following example shows output from the **show time** command.

```
console> show time
Mon Mar 31 2014 21:49:44 GMT+0200
```

show uptime

To view the system up time, use the **show uptime** command.

show uptime

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode and Enable mode

Examples The following example shows output from the **show uptime** command.

```
console> show uptime
50 days, 0h:00m:58s
```

show version

To view the installed software version for management and caching engine, use the **show version** command.

show version

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode and Enable mode

Examples The following example shows output from the **show version** command.

```
console> show version
show version
```

```
VDS-TC Transparent Caching           cli version 5.7.3b53
management           VDS-TC Transparent Caching mgmt software version 5.7.3b53
PANG                 5.7.3b53          LLPF Version LLPF_05.7.3b53-54
```

show volumes

To view a list of mounted volumes, use the **show volumes** command.

show volumes

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following example shows output from the **show volumes** command.

```
console> show volumes
Licensed volumes : 12
Volume name          State                Owner
/mnt/vol1            mounted              ce-1
/mnt/vol2            mounted              ce-1
/mnt/vol3            mounted              ce-1
/mnt/vol4            mounted              ce-1
/mnt/vol5            mounted              ce-1
/mnt/vol6            mounted              ce-1
/mnt/vol7            mounted              ce-1
/mnt/vol8            mounted              ce-1
/mnt/vol9            mounted              ce-1
/mnt/vol10           mounted              ce-1
/mnt/vol11           inactive
/mnt/vol12           inactive
```

For more information on available CLI monitoring commands, *see* Chapter 4, "Using CLI Commands".

To view status and statistics using SNMP, use any SNMP monitoring tool.

CLI Reference

The following is a tree structure of the CLI commands that are available in a VDS-TC Integrated Appliance configuration.

Regular Mode

```
arp                     Display ARP table
current_cli_users       Show currently logged in cli users
direction               Calculate the visible subnets on the interface
dmesg                   Display the dmesg
enable                  Enter Enable mode
eventlog                Event log commands
    date                    Display eventlog of a specific date
    export                  Export event log to TFTP server
    show                    Display event log
    tail                    Display online event log
exit                    Logs you out from the CLI
help                    Display CLI commands
ifconfig                Display the interface(s)
iostat                  Display extended I/O statistics
jumbo                   Send jumbo echo messages
ping                    Send echo messages
show                    Show run-time information
    detection_rules         Display detection rules
    dstat                   Display IO statistics
    eth_status              Display the eth status
    eventlog                Display event log
    leader                  Display current cluster leader
    process                 Display status of VDS TC components
    status                  Display cluster administrative and application status
    systemid                Display system serial number
    time                    Display system date and time
    uptime                  Display cluster uptime
    version                 Display software version
    vip                     Show VIP information
    volumes                 Display mounted volumes
traceroute              Display a packet's route
```

Enable Mode

Enable mode includes the commands available in Regular mode, and commands available only in Enable mode, as follows:

```
access                  Manage system access parameters
    enable-password        Enable mode password
    idle-session-timeout   Set idle session timeout value
    user-password          Regular mode password
apache_restart             Restart apache
arp                        Display ARP table
cache                      Cache operations
    active_sessions        Show active sessions for ip
    black_list             Manage the cache black list.
        add                Add a file to the black list using hash ID and protocol.
        dump               Display (dump) the entire black list.
        export             Export the black list to a TFTP server.
        remove             Remove a file from the black list using a hash ID and protocol.
    hash                   Display the file's metadata using a hash ID.
    list                   Display and exports the list of cache content.
        display            Display full list of cache content.
        export             Export cache content to TFTP server.
        short              Display the Least Recently Used cached HASH IDs.
    remove                 Remove a file from the cache using hash ID.
    summary                Display CMDB statistics summary.
    sync                   Verify and synchronize cache metadata
    volume                 Manage cache volumes
        activate           Activate a cache volume
        deactivate         Stop using a specific volume for caching
        remove_content     Erase volume's metadata
config                     Enter Configuration mode
current_cli_users          Show currently logged in cli users
detection_rules            Manage detection rules configuration
direction                  Calculate the visible subnets on the interface
dmesg                      Display the dmesg
eventlog                   Event log commands
    date                   Display eventlog of a specific date
    export                 Export event log to TFTP server
    forward                Starts event log forwarding
    show                   Display event log
    stop                   Stops event log forwarding
    tail                   Display online event log
exit                       Logs you out from the CLI
help                       Display CLI commands
ifconfig                   Display the interface(s)
iostat                     Display extended I/O statistics
jumbo                      Send jumbo echo messages
license                    Manage system license
    activate               Activate system license
    import                 Import license from TFTP server
    show                   Display current license
oper                       System management operations
    service                Manage services
        powerdown          Service power down
        powerup            Service power up
        start              Start VDS TC software and services
        stop               Stop VDS TC software and all its services
ping                       Send echo messages
reset                      Reset management services
show                       Show run-time information
    config                 Display running configuration
    detection_rules        Display detection rules
    dstat                  Display IO statistics
```

```
    eth_status            Display status of the ethernet interfaces
    eventlog              Display event log
    leader                Display current cluster leader
    license               Display system license information
    process               Display status of VDS TCVDS TC components
    status                Display cluster administrative and application status
    systemid              Display system serial number
    time                  Display system date and time
    uptime                Display cluster uptime
    version               Display software version
    volumes               Display mounted volumes
system                 System information
    logs export           Export the system logs to TFTP server
    statistics export     Export the system statistics to TFTP server
traceroute             Display a packet's route
upgrade                Upgrade VDS TC software version
vlan                      Display a list of users currently logged in
    add                   Add a vlan to an interface
    remove                Remove a vlan from an interface
```

Configuration Mode

```
apply          Apply config changes
diff           Show pending changes
discard        Discard pending changes
display        Display pending configuration
exit           Exit current mode
export         Export cluster configuration to TFTP server
help           Commands description
import         Import cluster configuration from TFTP server
network        Configure management network interface
    default_gw Configure default gateway
    ip         Configure management network interface
restore        Restore last good configuration
time           Set system date and time
```

PART 3

Cluster Configuration

Part 3 of the Cisco Videoscape Distribution Suite Transparent Caching Software Configuration Guide provides hands-on information and guidance for working with VDS TC Cluster configurations.

This part contains the following chapters:

Working with Cisco VDS TC Management Tools (Cluster)

This chapter describes how to use the management tools that come with Cisco Videoscape Distribution Suite Transparent Caching (VDS TC), including the CLI, file upload and download capabilities, the configuration file, and SNMP. The information provided in this chapter is required to perform the tasks explained in this guide.

VDS TC uses a number of different tools to help you configure, manage, and monitor its performance. Some management tasks allow you a choice of tools. For instance, you can use both the CLI and SNMP to view statistics.

Table 7-1 lists the different VDS TC management tools that are explained in this chapter.

Note You can also use the VDS TC Manager to manage the VDS TC caching engine. For information on managing the VDS TC caching engine using VDS TC Manager, see *Cisco Videoscape Distribution Suite Transparent Caching Manager User Guide*.

Table 7-1 Management Tools

Management Tool	Description
CLI	Used to perform the bulk of configuration and management tasks. For a complete description of VDS TC CLI commands, refer to Chapter 9, "Using CLI Commands (Cluster)".
TFTP	Cisco provides license and software upgrades using upgrade files that are downloaded to the VDS TC installation using a TFTP server. In addition, you can modify the configuration by downloading a configuration file using TFTP.
Configuration File	Used to configure operational modes, caching, and SNMP settings.
SNMP	Used to monitor or view VDS TC operational statistics. Note: Information available using SNMP is also available using the CLI.

Working with the CLI

You can use the CLI to perform configuration, management, and monitoring tasks, such as:

- Configure management settings, including:
  - Passwords
  - Management IP addresses
  - Local time
- Perform system maintenance, including:
  - Managing the caching service
  - Managing Servers
- Monitor the system. You can also use SNMP to monitor the system.
- Upgrade the system, including:
  - Upgrade the software
  - Upgrade licenses
  - Upgrading the system software or license is performed by downloading a new software or license file to the VDS TC device.

Note For a complete list of all CLI commands, refer to Chapter 9, "Using CLI Commands (Cluster)".

Getting Started with the CLI

There are two ways to access the VDS TC CLI:

- **Serial console:** The serial console is used to access the Regular CLI and the Rescue CLI. The Rescue CLI is used for entering the basic network and login information that you need to get the system up and running and is only available from the serial console.
  - To access the Regular CLI, log in as **admin**.
  - To access the Rescue CLI, log in as **rescue**.

  VDS TC ships with default passwords for both the admin and rescue users, which is a specific system serial number. You should change these passwords after installation. For instructions on changing passwords, *see* Chapter 8, "Managing Passwords".
- **LAN connection using SSH:** Use SSH to connect to the CLI over a LAN connection for regular access to the CLI. When accessing the CLI using a LAN connection, use the login **admin**.

Note Before configuring the network settings for VDS TC, or if you have changed your network settings so that VDS TC is not accessible from the outside, you must use the serial console connection.

Follow these steps to log into the CLI using the serial console:

Step 1 Connect to the USB port on the leftmost server in the chassis using a USB-to-Serial dongle. In the terminal emulator communications software (such as HyperTerminal or Tera Term) configure the following serial settings:

- Speed: 57600
- Data bits: 8
- Parity: None
- Stop bit: 1
- Software flow control disabled

Step 2 Press **Enter** and a login window appears.

Step 3 Enter the maintenance rescue login and press **Enter**.

Step 4 Enter the rescue user password and press **Enter**. Wait while the setup script completes and the system automatically logs out.

Follow these steps to log into the CLI using SSH over a LAN connection:

Step 1 Connect to VDS TC using SSH from anywhere on your LAN. A login prompt appears.

Step 2 Enter **admin** and press **Enter**.

Step 3 Enter the password and press **Enter**. The CLI prompt `console>` appears.

Once you are logged into the CLI using either the serial console or LAN connection, enter **help** or press **?** to view a list of commands.

To logout of the CLI when you are done, enter **exit**.

CLI Command Editing Features

You can edit CLI commands using the following keystrokes:

- Press the **TAB** key to auto-complete a command. If multiple choices are available, they are displayed, one option per line.
- Press **?** to display a command and its parameter hints. Enter as much of the command as you know and then press **?** to display the completion options and their descriptions. Each option is displayed one per line.
- After a command completes, you can display the next parameter hint by pressing **SPACEBAR+?**.
- Use the Up arrow and the Down arrow keys to navigate through the command history.

CLI Modes

VDS TC supports the following CLI modes:

- **Regular mode:** From this mode you can view system configuration and statistics, but you cannot change the settings. In Regular mode, the console> prompt appears and you can either enter Enable mode or exit the CLI.
- **Enable mode:** From this mode you can update the license or software, set the date, and configure the login name and password. In Enable mode, the console# prompt appears and you can enter configuration mode or return to Regular mode using the **exit** command.
- **Configuration mode:** From this mode you can configure any settings on the system. In Configuration mode the configuration# prompt appears and you can return to Enable mode using the **exit** command.

Note Configuration mode can only be used by a single user at a given time.

Caution If you exit Configuration mode without applying your changes, these changes are lost.

- **Server Mode:** Allows you to start, stop, or restart an individual server, and set the log level of the server. In Server mode the CLI server prompt appears: oper server <server number> # (for example: oper server 1#) You can return to Enable mode by entering the **exit** command.
- **Rescue CLI:** The Rescue CLI allows you to recover incorrect management network configurations and lost or forgotten CLI passwords. In Rescue CLI mode the rescue@ce-1# prompt appears and you can execute the following commands:
  - **access:** Resets the white and black management access lists.
  - **passwords:** Resets the admin and rescue passwords.
  - **network:** Configures the following network parameters: IP address, subnet mask, default gateway, and DNS server.
  - **exit:** Exits the Rescue CLI mode.
  - **help:** Displays a list of Rescue CLI commands. You can also display a list of Rescue CLI commands by pressing ? at the CLI rescue prompt.

After executing these commands, you can immediately perform another one or you can exit the Rescue CLI mode.

Switching from Regular Mode to Enable Mode

Follow these steps to switch from Regular mode to Enable mode:

Step 1 At the CLI prompt, enter **enable**. A password prompt appears.

Step 2 Enter the Enable mode password and press **Enter**. You are now logged in to Enable mode. The prompt should now show console#.

Note Your system serial number is the default password for the Enable mode. Ensure that you change it at installation. For instructions on how to change passwords, *see* Managing Passwords, page 8-28.

While logged into Enable mode you may need to run configuration commands. To run configuration commands you must enter Configuration mode. To enter Configuration mode from the Enable mode, enter **config**. You are now in Configuration mode.

To exit Configuration mode, enter **exit**. You are returned to Enable mode. If you enter **exit** in Enable mode, you are returned to Regular mode.

Switching to Server Mode

Follow these steps to switch from Regular mode to Enable mode:

Step 1 At the CLI enable prompt, enter **oper server** *server_number*, where *server_number* is the number of the server you want to control (for example: 1).

Step 2 The CLI server prompt appears: for example: oper server 1#

To exit Server mode, enter *exit*. You are returned to Enable mode. if you enter **exit** in Enable mode, you are returned to Regular mode.

TFTP Server

To configure VDS TC for uploading and downloading files, you must have an external TFTP server running. You download files to VDS TC using a TFTP server to do the following:

- Update the VDS TC software
- Update the VDS TC license
- Update the configuration by changing the configuration file

Before downloading a file to VDS TC, you must place the file on the TFTP server.

Note You can also upload files to VDS TC using VDS TC Manager. This eliminates the need to work with an external TFTP server.

Working with the Configuration Files

There are three configuration files that are used to configure different aspects of the VDS TC platform. Each file is responsible for different operational activities:

- The main configuration file is an XML file that manages operational modes, cache settings, and SNMP settings for VDS TC. This configuration file is an XML file. The common name for this file is cluster_conf.xml.
- Two additional configuration files define the traffic categorization rules: one file contains the major categories, also referred to as groups, and the other file contains the subcategories, also referred to as signatures. For example, a major category listed in the groups file might be Video Streaming, and its subcategories listed in the signatures file might be youtube, google.video, and video.facebook. These files are also XML files.

To modify the functionality of the VDS TC system, you can make changes to these configuration files directly and then upload the changes to the system.

Changing the Configuration Files

To make changes to the configuration files and apply these changes to the VDS TC system, follow these steps:

	Command	Purpose
Step 1	console# **config**	Enters Configuration mode.

	Command	Purpose
Step 2	`configuration# export TFTP_server filename`	Exports the current configuration file to a TFTP server. • The *TFTP_server* parameter is the IP address or hostname of the TFTP server to which you want to export the configuration file. To export the file to the /tftpboot directory on the VDS TC platform use **localhost** for this parameter. • The *filename* parameter is the name of the file to which you want to save the current configuration. The common filename to use is cluster_conf.xml.
Step 3	`configuration# exit`	Exits Configuration mode and returns to Enable mode.
Step 4	`console# detection_rules export_groups TFTP_server filename`	Exports the major categories (groups) to a TFTP server. • The *TFTP_server* parameter is the IP address or hostname of the TFTP server to which you want to export the groups file. To export the file to the /tftpboot directory on the VDS TC platform use **localhost** for this parameter. • The *filename* parameter is the name of the file to which you want to save the major categories.
Step 5	`console# detection_rules export_signatures TFTP_server filename`	Exports the subcategories (signatures) to a TFTP server. • The *TFTP_server* parameter is the IP address or hostname of the TFTP server to which you want to export the signatures file. To export the file to the /tftpboot directory on the VDS TC platform use **localhost** for this parameter. • The *filename* parameter is the name of the file to which you want to save the subcategories.
Step 6	Open the configuration file in a text editor or an XML editor.	Make the desired changes to the configuration files and save the changes. **Note** When editing the configuration files, edit only the field values and do not change or erase the XML markup tags. If XML tags are changed, the configuration will be rejected upon loading.
Step 7	`console# config`	Enters Configuration mode.
Step 8	`configuration# import TFTP_server filename`	Imports the new configuration file to VDS TC. • The *TFTP_server* parameter is the IP address or hostname of the TFTP server from which you are importing the configuration file. To import the file from the /tftpboot directory on the VDS TC platform use **localhost** for this parameter. • The *filename* parameter is the name of the configuration file that you want to import.

	Command	Purpose
Step 9	`configuration# display`	Displays the currently loaded configuration. This configuration will not take effect until you enter the **apply** command. To view the differences between the currently active configuration and the newly imported configuration, use the **diff** command. The differences are marked with a plus (+) and a minus (-): • **minus (-):** Represents the exiting configuration, prior to the import. • **plus (+):** Represents the new configuration, after the import.
Step 10	`configuration# apply`	Applies the new configuration.
Step 11	`configuration# display`	Displays the current configuration.
Step 12	`configuration# diff`	Shows pending changes. The differences between the currently active configuration and the newly imported configuration are marked with a plus (+) and a minus (-): • **minus (-):** Represents the exiting configuration, prior to the import. • **plus (+):** Represents the new configuration, after the import. If you have successfully applied the new configuration, this command should display “Configurations are identical”.

Note To discard the pending changes, use the **discard** command in Configuration mode. If you have applied the changes and want to revert back to the last known good configuration, use the **restore** command in Configuration mode.

Configuration File Sections

The main configuration file has three main sections:

- **mgmt-config:** Use this section to configure the network settings on the system, such as the management IP address and default gateway.
- **common:** Use this section to define the default settings for all servers. Settings in the individual server sections override the fields in this section.
- **blade id=#:** # represents the number of the caching engine server (slot number minus 1). Use this section to define the settings for an individual server. Some of the fields in the <service> subsection of the <blade id> section are identical to fields in the <service> subsection of the <common> section. When there is a value for a field in both sections, the value of the field in the <blade id> section takes priority over the value of the field in the <common> section.

Accessing VDS TC SNMP Information

VDS TC provides a robust set of SNMP status information that you can monitor using any standard SNMP tool. All status information that is available using SNMP is also available using status commands at the CLI. The SNMP information is provided using a private MIB (SNMP v2) environment. The private MIB file is located in the /opt/pang/mgmt/avalon/share/snmp/mibs folder on the VDS TC Management Server and is named VDS-TC-MIB.txt.

Configuring VDS TC

This chapter describes the steps necessary to configure and perform maintenance on the VDS TC system if changes are required on a running platform. This chapter has the following sections:

Main Operational Features (Quick Jumpstart)

You must initially configure the following elements to jumpstart the system. Detailed configuration instructions are provided later in this chapter.

- CLI-based configuration:
  - Configuring the management network
- File-based configuration:
  - Configuring SNMP
  - Configuring P2P protocols
  - Configuring traffic forwarding options: When you deploy a VDS TC Cluster solution, there are several different supported L7 device configurations that can be used. Symmetric/asymmetric connection modes with single/multiple port connections are possible when you are configuring L7 devices with a VDS TC Integrated Appliance platform.

VDS TC Features

The following is a description of system features that are available in VDS TC. You can configure some of these features using the VDS TC configuration file and some of them using the VDS TC CLI.

Note Any changes that you make to the VDS TC configuration file will only take affect after you import the configuration file into the VDS TC management server. For information on how to import this file, see Working with the Configuration Files in Chapter 7, "Working with Cisco VDS TC Management Tools (Cluster)".

Caching Specific Features

- **Black list of hashes:** You can black-list specific hashes as non-cacheable which prevents the hashes from being cached or provided again to users, if already cached. The black list is maintained using the CLI, allowing administrators to add, remove, or view the list of hash IDs in the black list. *See* the cache command, page 9-22.
- **Selective caching:** The VDS-TC platform includes a dynamic mechanism that automatically decides if a specific large content item should be cached-in as popular content. Based on this decision, made for each content item that is requested by a user, the content is either cached-in or forwarded to the user. The platform allows the operator to disable this mechanism. If you disable this feature, all content items that are requested by a user are immediately cached-in and ignore any "popularity-algorithm" decision making.

  Setting the <selective_cache_in_threshhold> parameter to 0 disables the selective caching feature and forces the caching-in of all large content items, ignoring the popularity that is associated with these files. To disable the selective caching feature enter the following configuration in the <policy> subsection of the <service> section:

  ```
  <selective_cache_in_threshhold>0</selective_cache_in_threshhold>
  ```

  Note Disabling the selective caching feature on a production platform will lead to caching all content items that are requested by users. This is not recommended because the storage will fill up rapidly and the cache-out performance will be dramatically decreased.

- **Small memory buffer:** When traffic is very low because of a shaper configuration, the memory buffer size might be too large (in memory) for this shaper. This feature allows you to control the buffer size, to optimize memory use and tailor it to the way the traffic is shaped. If the buffer size does not match the shaped traffic size, the cache will fill up too slowly. You can configure the number of buffers in the configuration file.

  To control the number of buffers that are used, add or edit the following text in the <service> section in the configuration file where the *number* parameter is the number of buffers that you would like to use, such as 8000:

  ```
  <memory>
          <small_io_blocks>number</small_io_blocks>
  </memory>
  ```

- **Bandwidth-per-connection management:** This configuration option controls the cache-out sessions (bandwidth - management), which enables you to place a top limit on the cache-out sessions.

  To modify the cache-out sessions, add or edit the following text in the <service> section of the configuration file, where the *max_bw_per_IP* parameter is the maximum bandwidth per IP in b/s:

  ```
  <bandwidth-management>
      <enable-bandwidth-management>1<enable-bandwidth-management>
      <bandwidth-per-connection>max_bw_per_IP<bandwidth-per-connection>
  </bandwidth-management>
  ```

- **Administrative state locked:** This configuration option prevents the server on which it is configured from handling traffic. The traffic is not processed.

  To lock the server, add the following text to the configuration file in the <blade id=#> section:

```
<cache-engine>
    <administrative_state>locked</administrative_state>
</cache-engine>
```

- **Upstream caching:** A last mile architecture suffers from limited upstream resources that are gravely affected from peer-to-peer symmetrical traffic pattern. Upstream caching relieves network congestion by providing cached pieces to peers in other "zones", directly from the cache instead of the last mile user. There is no configuration for this feature.
- **HTTP caching:** You can configure VDS TC to cache HTTP documents (such as video files, video streaming, and images) to reduce bandwidth usage and to improve the user experience by providing accelerated document download time. When you implement transparent HTTP caching, any standard HTTP contained document can be cached regardless of the URL that is associated with it.

  To enable VDS TC to support the HTTP protocol, add the following text to the configuration file in the <protocols> subsection of the <service> section:

```
<enable-http>1</enable-http>
```

- **HTTP minimum file size:** You can modify the minimum size a file needs to be in order to be cached by the system. Files requested using HTTP that are larger than the value defined with the http_min_file_size parameter are cached by the system, if popular. Files that are smaller than this value are not cached by the system.

  The default value for the http_min_file_size parameter is 512 Kb. To change this value, add the following text in the <policy> subsection of the <service> section in the configuration file, where the *file_size* parameter is the minimum file size in bytes:

```
<http_min_file_size>file size</http_min_file_size>
```

  For example:

```
<http_min_file_size>65536</http_min_file_size>
```

Note It is recommended that you only change the HTTP minimum file size at the default of 512 Kb if instructed to by your Cisco Support System Engineer.

- **Selective HTTP caching:** The VDS-TC platform includes a dynamic mechanism that automatically decides if a specific large HTTP item should be cached-in as popular content. Based on this decision, made for each content item that is requested by a user, the content is either cached-in or forwarded to the user. The platform allows the operator to disable this mechanism. If you disable this feature, all content items that are requested by a user are immediately cached-in and ignore any "popularity-algorithm" decision making.

Note Disabling this dynamic caching selection method is only appropriate in lab environments.

To disable the dynamic caching selection method so that the system starts caching the HTTP requests on the first request, add the following text in the <policy> subsection of the <service> section in the configuration file.

```
<http_selective_cache_in_threshhold>0</http_selective_cache_in_threshhold>
```

- **Flush hours:** The max_hours_hash_not_touched parameter sets the amount of time to wait, in hours, before flushing an empty HASHID (which is created when seeing a request for the file for the first time) from the cache index if the file is empty. The default value is 24 hours.

Note This parameter is for HTTP only.

To change this value, add or edit the following text in the <policy> subsection of the <service> section in the configuration file:

<max_hours_hash_not_touched>*hours***</max_hours_hash_not_touched>**

For example:

```
<max_hours_hash_not_touched>48</max_hours_hash_not_touched>
```

- **Ares protocol support:** Ares Galaxy is an open source P2P file sharing application and protocol that uses its own decentralized supernode/leaf network.

  To enable the VDS TC system to support the Ares protocol, add the following text to the configuration file in the <protocols> subsection of the <service> section:

  <enable-ares>1</enable-ares>

- **Do not cache specific URLs, hosts, or subnets:** You can configure VDS TC to avoid caching specific URLs, hosts, or subnets. Each URL seen in a GET request that floats through the VDS TC engine is compared to all of the URLs, hosts, and subnet entries in the <no_cache_url_list> section. When a match is found, the cache-in and cache-out are skipped and the request is forwarded as is. The configuration to avoid caching specific URLs, hosts, or subnets is configured in the <policy> subsection of the <service> section in the configuration file.

Note The <no_cache_url_list> section is limited to 64 entries. Use this option with caution. It imposes an extra burden on the VDS TC system because it compares all of the entries in the table with each URL and session details floating through the system.

  – The following is an example of how to add a URL to the no cache list:

```
<policy>
<no_cache_url_list>
    <url_no_cache>video_id</url_no_cache>
    <url_no_cache>videoplayback</url_no_cache>
</no_cache_url_list>
</policy>
```

  This example will match www.thegame.com/video_id/movie=8979 and www.thegame.com/video_id/movie=349587?speed=4. This example will *not* match www.thegame.com/playvideo/video_id=89779 because the video_id pattern does not match the beginning of the URI.

  – The following is an example of how to add a host to the no cache list, using either a hostname or a host IP address:

```
<policy>
<no_cache_host_list>
    <host_no_cache>shop.offlineshoppppping.com</host_no_cache>
    <host_no_cache>202.202.1.16</host_no_cache>
</no_cache_host_list>
</policy>
```

The host_no_cache option supports wildcard configurations for both a hostname and a host IP address. The following example will match any host on the offlineshopping.com, any host that has a name with "offlineshopping.com" in the name, and any host that has an IP address that matches either 202.202.1.* or *.202.202.1.

```
<policy>
<no_cache_host_list>
    <host_no_cache>offlineshopping.com</host_no_cache>
    <host_no_cache>202.202.1</host_no_cache>
</no_cache_host_list>
</policy>
```

This example will match the following:

aaa.offlineshopping.com

123456offlineshopping.com

jjj.offlineshoping.com.au

122.202.202.1

202.202.1.3

– The following is an example of how to add a subnet to the no cache list. The <no_cache_subnet_list> section matches any IP address that is associated with either the client requesting the content or the server servicing the request. Therefore, if one of the IP addresses that is associated with a specific session falls into the subnet IP range specified in this policy, the information will not be cached-in or cached-out:

```
<policy>
<no_cache_subnet_list>
    <subnet_no_cache>192.168.0.150</subnet_no_cache>
    <subnet_no_cache>192.168.1.0/24</subnet_no_cache>
    <subnet_no_cache>192.168.2.150-
        192.168.2.158</subnet_no_cache>
</no_cache_subnet_list >
</policy>
```

- **Cache data expiration:** There are several system parameters that affect cache data expiration:
  - **expiration_high_water_mark:** This parameter determines how full the cache data storage can become, based on percentage, before the system performs an expiration task. By default, the system checks this value at 0400 system time to determine if an expiration task needs to be performed. If the cache disk volume usage is above this parameter, the expiration task begins to run. The default value for this parameter is 97%.
  - **expiration_low_water_mark:** This parameter controls at what point the system expiration task can stop removing objects from the cache data storage. Once the cache disk volume usage falls below this parameter, the expiration task stops running. The default value for this parameter is 85%.
  - **volume_critic_water_mark:** When the cache disk volume usage reaches the level that is defined by this parameter, as a percentage of disk space used, the application stops writing to this volume. The default value for this parameter is 98%.
  - **Busy window:** The busy window determines the time frame during which the expiration task does not run. At the end of the busy window, the system checks the expiration_high_water_mark parameter to see if an expiration task needs to be performed.
    - **start_hour_for_busy_window:** This parameter determines the start time for the busy window. The default value for this parameter is 16 (1600 system time).

–**busy_window_size_in_hours:** This parameter determines the length of the busy window. You add this number of hours to the start_hour_for_busy_window to determine when the busy window has ended. The default value for this parameter is 12.

Based on the default values of the start_hour_for_busy_window and the busy_window_size_in_hours parameters, at 0400 system time the system will determine whether the cache data storage is at or above the expiration_high_water_mark parameter. If the cache data storage is at or above the high water mark, then the expiration task determines the least recently hit (least popular) object in the storage, and removes it. After this removal, the task checks to see if the expiration_low_water_mark parameter has been reached. If the low water mark has *not* been reached, the expiration task continues to remove the least popular objects until the low water mark is reached.

The cache data expiration process is performed for objects that have been cached-in. For objects that have gone through the VERIFY step, but are not yet cached-in (because they have been seen only once), the system will remove any record of hashes that have an age greater than that configured in the max_hours_hash_not_touched system parameter, for which the default is 24 hours. For more information on the max_hours_hash_not_touched system parameter, see the “Flush Hours” topic.

To modify the cache data expiration parameters, add or edit the following text to the <service> subsection of the <common> section:

```
<expiration>
   <expiration_low_water_mark>low_water_mark_%</expiration_low_water_mark>
   <expiration_high_water_mark>high_water_mark_%</expiration_high_water_mark>
   <volume_critic_water_mark>volume_critical_water_mark_%</volume_critic_water_mark>
   <start_hour_for_busy_window>start_time_for_busy_window</start_hour_for_busy_window>
   <busy_window_size_in_hours>number_of_hours</busy_window_size_in_hours>
</expiration>
```

For example:

```
<expiration>
    <expiration_low_water_mark>85</expiration_low_water_mark>
    <expiration_high_water_mark>92</expiration_high_water_mark>
    <volume_critic_water_mark>96</volume_critic_water_mark>
    <start_hour_for_busy_window>1400</start_hour_for_busy_window>
    <busy_window_size_in_hours>12</busy_window_size_in_hours>
</expiration>
```

Supporting Netflix

The Netflix protocol works differently on different devices. The supported devices in VDS TC are divided into two groups:

- iOS Apple devices, such as iPhones, IPads, and Apple TV
- Android OS devices, such as tablets, Smart Phones and Smart TVs, and Streamers

Configure the following to enable the VDS TC system to support the Netflix protocol:

- Add the following text to the configuration file in the <protocols> subsection of the <service> section:

```
<enable-netflix>1</enable-netflix>
<enable-silverlight>1</enable-silverlight>
```

- Add the following text to the <policy> subsection of the <service> section in the configuration file to configure support for the Netflix CDN IP ranges, where *CDN_IP_Range* is a CDN IP range that is appropriate for *your* installation.

```
<netflix_cdn_subnet_list>
    <netflix_cdn_subnet>CDN_IP_Range</netflix_cdn_subnet>
</netflix_cdn_subnet_list>
```

Note To add more than one CDN IP Range, add additional **<netflix_cdn_subnet>***CDN_IP_Range***</netflix_cdn_subnet>** lines.

For example:

```
<netflix_cdn_subnet_list>
<netflix_cdn_subnet>108.175.0.0/16</netflix_cdn_subnet>
<netflix_cdn_subnet>198.45.0.0/16</netflix_cdn_subnet>
</netflix_cdn_subnet_list>
```

Supporting Video Skips URL Strings Configuration

Cisco VDS TC handles video transactions as skips (jumps) if the URL includes some specific strings, such as: "start=, "begin=" that are found in the popular OTT video sites. Starting with Cisco VDS TC Release 5.2.0, you can add additional video skip/jump strings to match in addition to the default strings the application is seeking ("start=, "begin="). You should add only new string matches. You should not add the already existing default ones of "start=", "begin=".

You configure add additional video skip/jump strings within the <policy> subsection of the <service> section in the configuration file by adding the following configuration:

Note If there are additional parameters in the <policy> section, the <jump_string_match_list> should be the *first* element in this section.

```
<policy>
<jump_string_match_list>
<jump_string_match>relative_pos=#</jump_string_match>
</jump_string_match_list>
</policy>
```

This parameter and number represent a "jump" request in the URL, for example: seek_sec=61.

Note The <jump_string_match_list> structure can include up to 64 entries.

In addition, add the following value to the cluster_conf <policy> section:

```
<http_closed_file_timeout>25</http_closed_file_timeout>
```

System Load Monitoring

System load monitoring measures packet delays and packet loss, and if the packet delays or packet loss reach thresholds that you have configured, you can send an SNMP trap message or disable the service. You configure system load monitoring within the <policy> subsection of the <service> section in the configuration file.

- **check_overload_interval:** This field defines how frequently, in seconds, the NICs are polled for packet drops and packet delay.

- **overload_drop_percent:** This field defines a packet drop percentage threshold to monitor for, calculated per interface. If this threshold is exceeded a consecutive number of times, as defined in the <failed_overload_test> field, the overload action defined in the <overload_action> field occurs.
- **overload_packet_delay:** This field defines a packet delay threshold, in milliseconds, to monitor for, calculated across all interfaces. If this threshold is exceeded a consecutive number of times, as defined in the <failed_overload_test> field, the overload action defined in the <overload_action> field occurs.
- **failed_overload_test:** This field defines the number of consecutive times the packet drop threshold or the packet delay threshold must exceed their configured values to trigger the action defined in the <overload_action> field.
- **overload_action:** This field defines the action to take if the packet drop threshold or packet delay threshold has been exceeded for the configured number of consecutive times. This value can be NOTHING, TRAP_ONLY, or DISABLE.

The following example will poll interfaces every 12 seconds. If in three consecutive polls the delay is more than 500 ms or if packet loss is more than 1.22%, the service is disabled:

```
<policy>
    <check_overload_interval>12</check_overload_interval>
    <overload_drop_percent>1.22</overload_drop_percent>
    <overload_packet_delay>500</overload_packet_delay>
    <failed_overload_test>3</failed_overload_test>
    <overload_action>DISABLE</overload_action>
</policy>
```

Platform Specific Features

- **Cluster File System (CFS):** CFS is a distributed file-system that can operate seamlessly over *n* times storage devices. This provides a very large storage for each cache engine, enabling very fast data retrieval of cached data. The CFS is a content aware file-system, optimized specifically for the content it stores. It uses less I/O operations to service the amount of cached information it serves. Faster data throughput is achieved.

Platform Operational Specific Features

- **Configuring an NTP Server and Time Zone:** Follow these steps to configure the VDS TC management server to use an NTP server and configure the time zone for the VDS TC system:

  Add the following text to the configuration file in the <common> section:

  ```
  <ntp>
      <server-ip>ntp-server1-ip</server-ip>
      <server-ip>ntp-server2-ip</server-ip>
      <timezone>GMT <+/->offset</timezone>
  </ntp>
  ```

  where

  – *ntp-server1-ip* is the IP address or hostname of the first NTP server to use

Note To use the local server as the NTP server, enter **127.127.1.0** for the IP address.

- *ntp-server2-ip* is the IP address or hostname of a second NTP server to use, if desired
- *offset* is the GMT offset (+/-), for the time zone of the VDS TC system.

For example:

```
<ntp>
    <server-ip>1.asia.pool.ntp.org</server-ip>
    <server-ip>2.asia.pool.ntp.org</server-ip>
    <timezone>GMT+3</timezone>
</ntp>
```

Note The GMT *offset* value only supports a whole number between 0 and 12. All other time zones are NOT supported.

Note The time will not automatically adjust for daylight savings time (DST). If you need to adjust the time for DST, you will need to make this change manually.

Note If you upgrade to VDS TC 5.7.3, you may need to reconfigure the NTP settings.

Note When you use the **date** command from the VDS TC manager operating system, you may see that the GMT offset appears opposite of what you entered. For example if you entered **<timezone>GMT+8</timezone>** in the configuration file, when you enter the **date** command in the operating system, you will see "Tue August 1 14:16:35 GMT-8 2014". The output in the date command is intentionally reversed for backwards compatibility with POSIX standards.

- **Configuring Volume Distribution after a Cache Engine Failure:** This setting controls when the volumes of a failed cache engine will be redistributed (mounted) to the remaining cache engines. By default this value is 24 hours, which means that by default, if a cache engine fails, its volumes will not be redistributed (mounted) to the remaining cache engines until 24 hours after the cache engine fails.

  It is recommended that you leave the default to 24 hours, however if you need to change this setting, add the following text to the configuration file in the <service> section under the <memory> subsection. The *seconds* parameters is the number of seconds that the system should wait after a cache engine fails before its volumes are redistributed (mounted) to the remaining cache engines:

  ```
  <io>
      <volume_selection_algorithm_time>seconds</volume_selection_algorithm_time>
  </io>
  ```

  For example, to have the system wait 30 minutes after a cache engine fails before redistributing the volumes of the failed cache engine enter the following:

  ```
  <io>
      <volume_selection_algorithm_time>1800</volume_selection_algorithm_time>
  </io>
  ```

- **Duplicate Logs to External Syslog:** This feature duplicates the local syslog information of the VDS TC system and sends it to an external syslog server while the system is running.

  To enable this feature add the following text to the configuration file, where the *IP_address* parameter is the IP address of the external syslog server:

```
<mgmt-config>
    <external_syslog_ip>IP_address</external_syslog_ip>
</mgmt-config>
```

You can also stop and start forwarding the syslog to an external syslog server using the CLI. *See* eventlog, page 9-6.

- **Enabling Hardware Traps:** Enabling hardware traps enables you to see hardware traps in Cisco VDS Transparent Caching Manager. The equipment traps must also be configured in the target device. By default this feature is disabled.

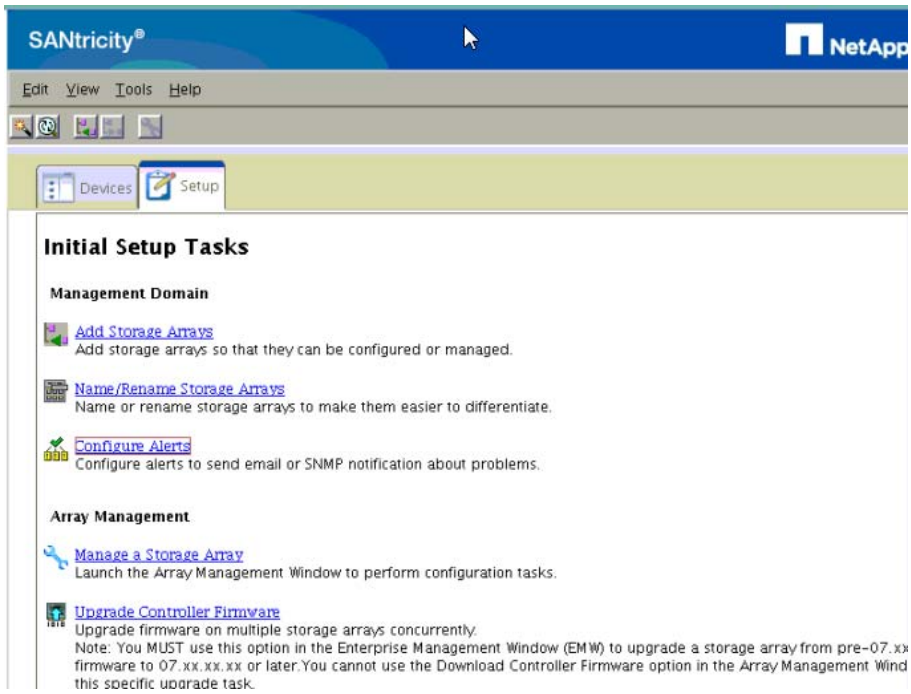
  To enable hardware traps, add the following text to the <mgmt-config> section of the configuration file:

  ```
  <enable-equipment-traps>1</enable-equipment-traps>
  ```

Note It is recommended that you configure the Cisco UCS Blade server to send traps to the NMS. However, if you prefer to configure the UCS Blade Server to send traps to the VDS TC management server, follow the steps that are available in the "SNMP Configuration Steps" section in the *UCMS Faults Report and SNMP Traps* document available at http://www.cisco.com/c/en/us/support/docs/servers-unified-computing/ucs-manager/112003-ucsm-faults-report-snmp-traps-00.html#steps.

- **Defining the Storage Traps on NetApp SAN:** To define the storage traps on the NetApp E2724, follow these steps:
  a. Open a VNC client connection to the VDS TC Management Server.
  b. Enter the command **/opt/SMgr/client/SMclient** to start the SANtricity Storage Manager software.
  c. From the Storage Manager client, click the **Setup** tab.
  d. From the Initial Setup Tasks window, click **Configure Alerts**.

Figure 8-1Configure Alerts



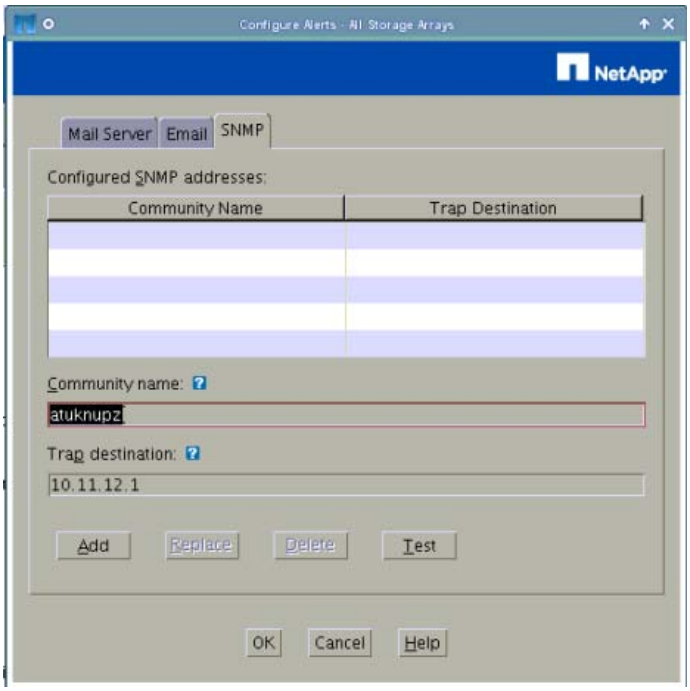
e. The Configure Alerts window appears. From this window, choose **All Storages** and click the **SNMP** tab.

f. In the Community Name field, enter the community name to use.

g. In the Trap Destination field, enter the IP address of the SNMP trap server, which is the Management Server. This address should be 10.11.12.1.

h. Click **Add** to add this configuration and click **OK** to close the Configure Alerts window.

Figure 8-2 Configure Alerts Pop-Up



i. Close the NetApp Storage Manager client.

- **Defining the Storage Traps on IBM SAN:** To define the storage traps on the IBM DS3524 NAS, follow these steps:

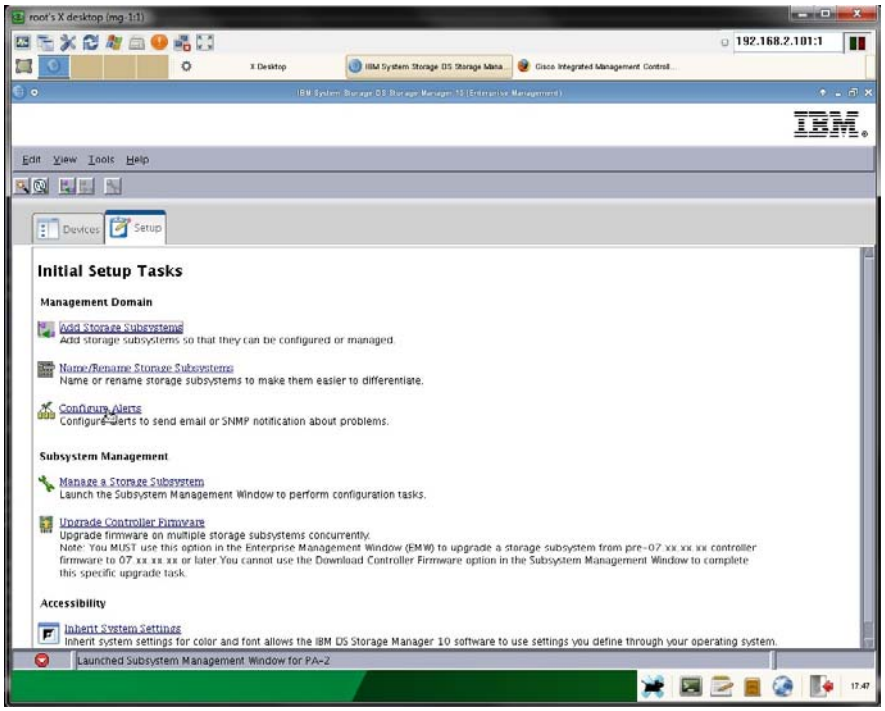
a. Open a VNC client connection to the VDS TC Management Server.

b. Enter the command **/opt/IBM_DS/client/SMclient** to start the IBM Storage Manager software.

c. From the Storage Manager client, click the **Setup** tab.

d. From the Initial Setup Tasks window, click **Configure Alerts**.

Figure 8-3Configure Alerts



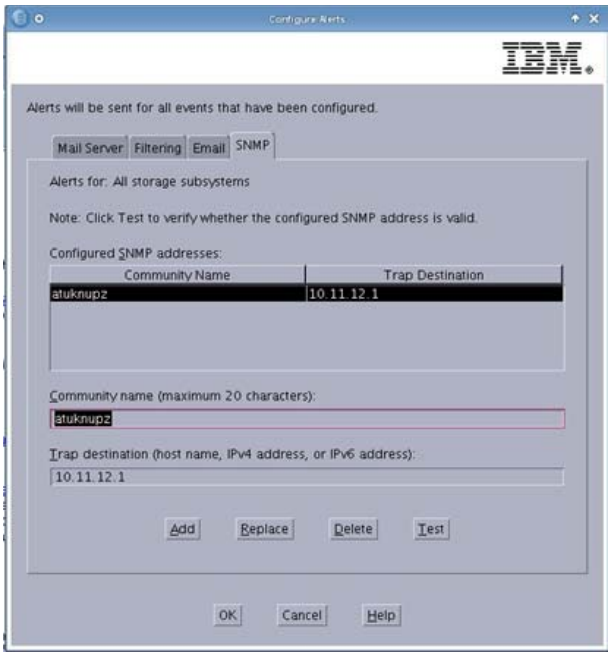
e. The Configure Alerts window appears. From this window, choose **All Storages** and click the **SNMP** tab.

f. In the Community Name field, enter the community name to use.

g. In the Trap Destination field, enter the IP address of the SNMP trap server, which is the Management Server. This address should be 10.11.12.1.

h. Click **Add** to add this configuration and click **OK** to close the Configure Alerts window.

Figure 8-4Configure Alerts Pop-Up



i. Close the IBM Storage Manager client.

- **Management Access Control List (ACL):** This feature enables you to configure a white list or a black list to determine what source IP addresses are allowed to access the VDS TC management interfaces.
  - **White list:** If you use a white list, enter the IP addresses that should be permitted to access the VDS TC management interfaces. Any IP addresses that do not match an entry in the white list are denied access to the management interfaces.
  - **Black list:** If you use a black list, enter the IP addresses the you do not want to have access to the VDS TC management interfaces. Any IP addresses that do not match an entry in the black list are permitted access to the management interfaces.

  To use a white list to determine who should be allowed access to the VDS TC management interfaces, add the following text to the <mgmt-config> section of the configuration file, just *before* the <TACACS_configuration> section, if present. The *permitted_IP_address* parameter is the IP address of a host that should be permitted access to the VDS TC management interfaces:

```
<access_lists>
    <white_access_list>
        <access_entry>permitted_IP_address</access_entry>
        <access_entry>permitted_IP_address</access_entry>
    </white_access_list>
</access_lists>
```

  For example:

```
<access_lists>
    <white_access_list>
        <access_entry>192.168.1.1</access_entry>
        <access_entry>192.168.1.2</access_entry>
    </white_access_list>
</access_lists>
```

Note Any IP addresses that do not match an entry in the white list are implicitly denied.

To use a black list to determine who should be allowed access to the VDS TC management interfaces, add the following text to the <mgmt-config> section of the configuration file, where the *denied_IP_address* parameter is the IP address of a host that should be denied access to the VDS TC management interface:

```
<access_lists>
    <black_access list>
        <access_entry>denied_IP_address</access_entry>
        <access_entry>denied_IP_address</access_entry>
    </black_access list>
</access_lists>
```

For example:

```
<access_lists>
    <black_access list>
        <access_entry>80.122.12.1</access_entry>
        <access_entry>80.122.12.2</access_entry>
    </black_access list>
</access_lists>
```

Note Any IP addresses that do not match an entry in the black list are implicitly permitted.

- **Forwarding SNMP traps for the VDS TC caching service and CIMC:** The VDS TC caching service generates SNMP traps for certain events, and you can configure the system to forward these traps to an external server.

  Follow these steps to control the forwarding of VDS TC traps:

Step 1 Obtain an SNMP monitoring system.

Step 2 Add the following text to the configuration file in the <common> section:

```
<snmp>
<trap-ip>IP_Address_SNMP_Server</trap-ip>
<snmp-read-community>read_community_string</snmp-read-community>
<snmp-write-community>write_community_string</snmp-write-community>
<snmp-trap-community>trap_community_string</snmp-trap-community>
</snmp>
```

For example:

```
<snmp>
    <trap-ip>10.11.12.1</trap-ip>
    <snmp-read-community>gdcbhv</snmp-read-community>
    <snmp-write-community>nkppui</snmp-write-community>
    <snmp-trap-community>ffff</snmp-trap-community>
</snmp>
```

- **Email alerts:** In addition to SNMP traps, you can forward critical VDS TC alerts to a specific email server and email address.

To configure the forwarding of critical alerts to an email address, add the following text to the configuration file where the *dns_server* parameter points to a DNS server that can resolve the domain name to which the email should be sent:

```
<mgmt-config>
    <nameserver>dns_server</nameserver>
    <alert-email>email_address</alert-email>
</mgmt-config>
```

For example:

```
<mgmt-config>
    <nameserver>194.90.1.5</nameserver>
    <alert-email>support@cisco.com</alert-email>
</mgmt-config>
```

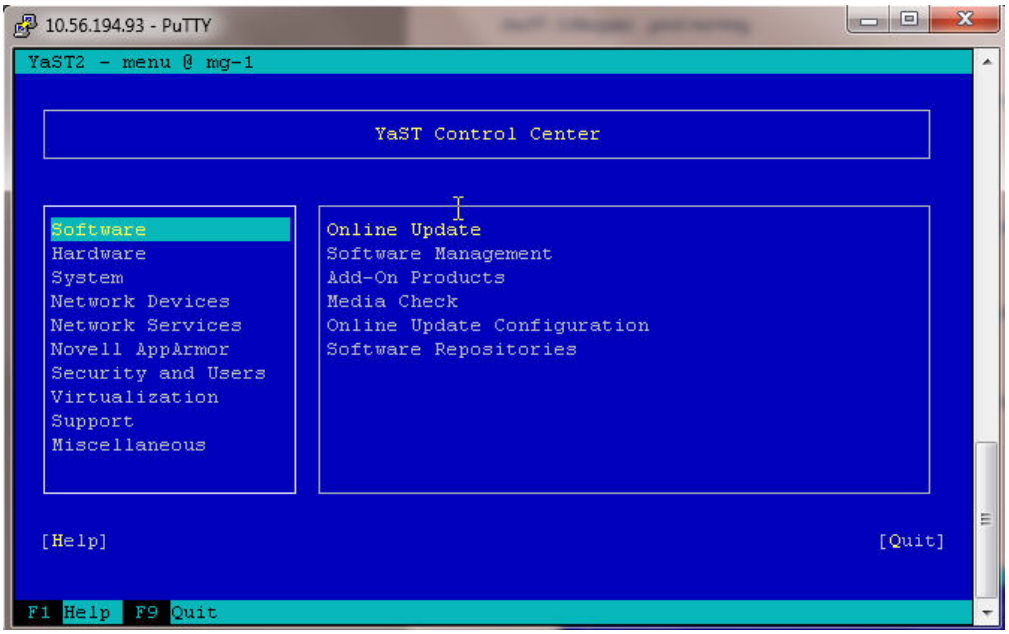
Note If the VDS TC Manager needs to use an SMTP e-mail relay, you must also perform the following steps:

Step 1 Using SSH software, log into the VDS TC management server using the username **padmin** and the password that was provided by Cisco.

Step 2 Enter the **su root** command to switch to the root user. Enter the password that was provided by Cisco.

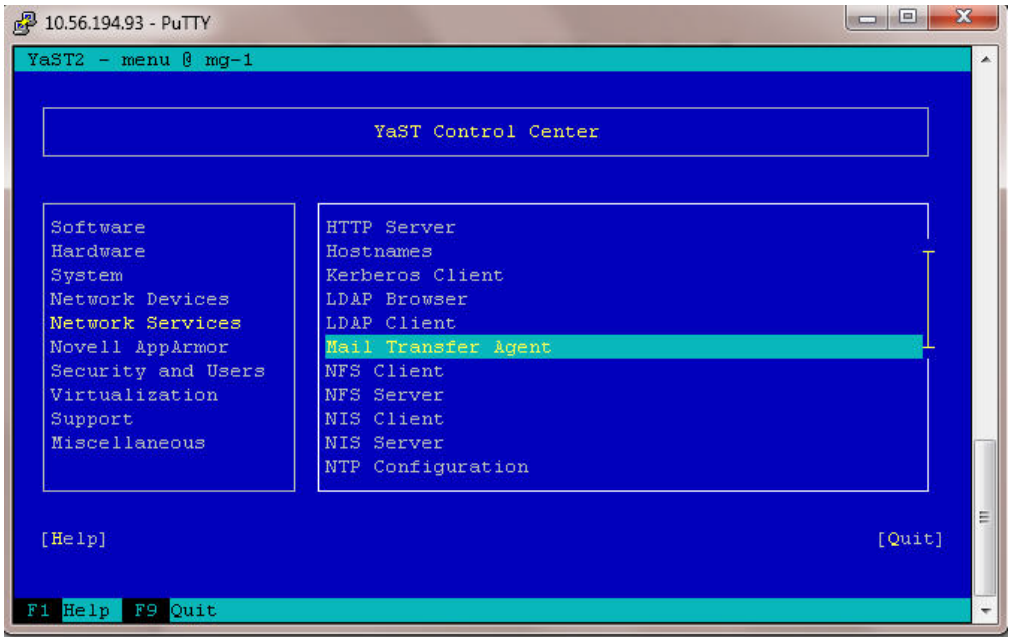
Step 3 Enter the command **yast** to enter the YaST environment.

Figure 8-5 YaST Control Center



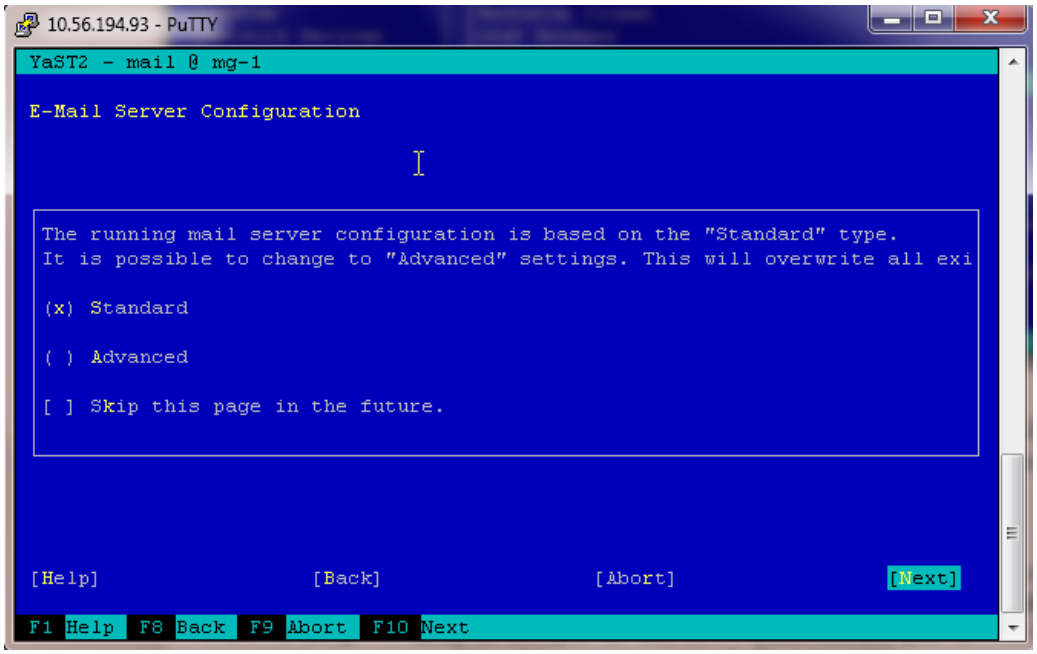
Step 4 From the YaST Control Center window choose **Network Services** from the left pane and then choose **Mail Transfer Agent** from the right pane. Press **Enter**.

Figure 8-6 Network Services - Mail Transfer Agent



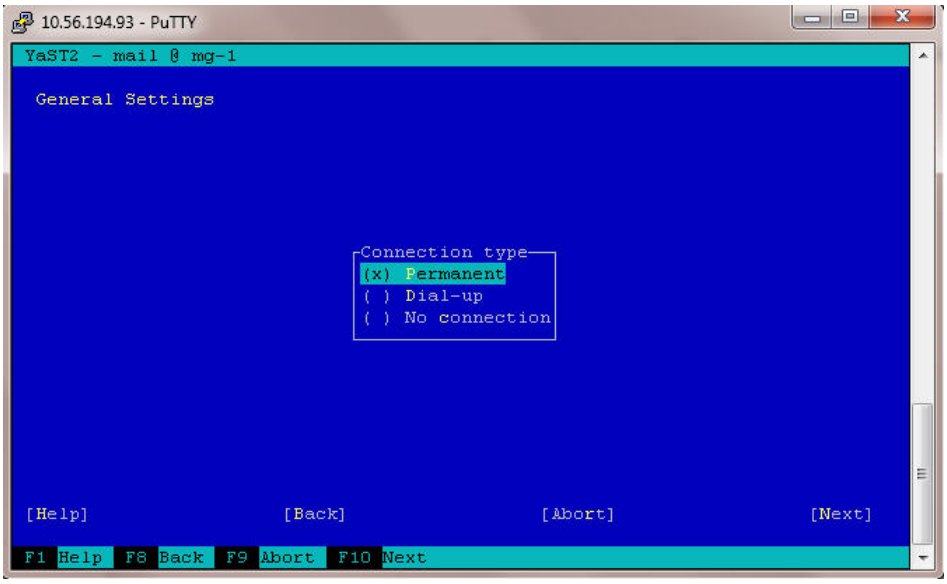
Step 5 The E-Mail Server Configuration window appears. From this window if Standard is not selected (an X will appear in front of the option if it is selected), press **Alt-S** to select it and then press **Alt-N** to proceed to the next window.

Figure 8-7 E-mail Server Configuration



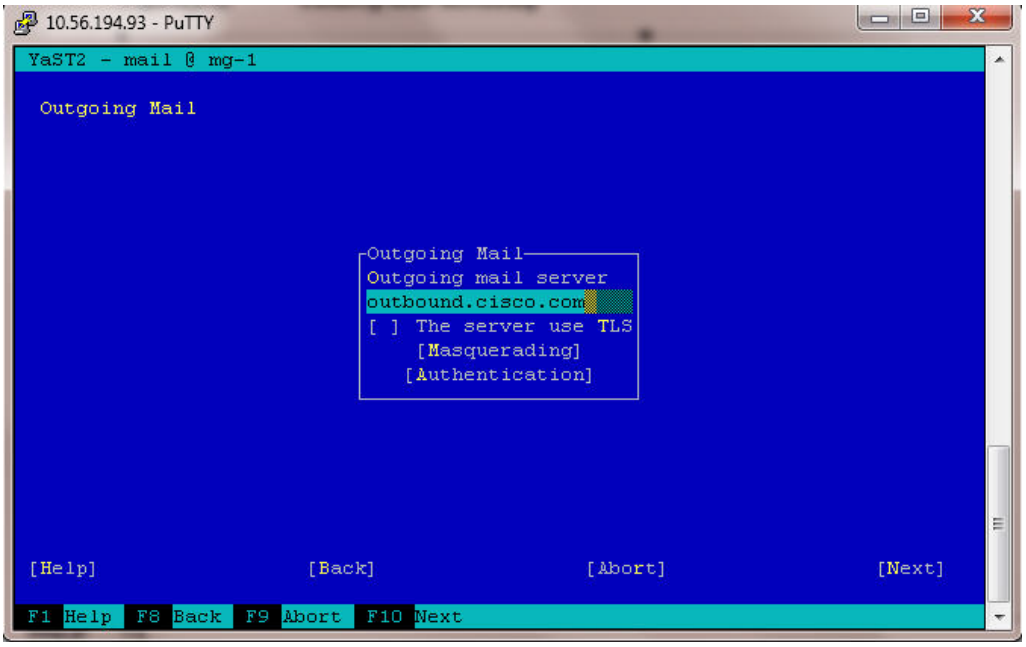
Step 6 The General Settings window appears. From the Connection Type section, if Permanent is not selected, press **Alt-P** to select it and then press **Alt-N** to go to the next window.

Figure 8-8 General Settings Window



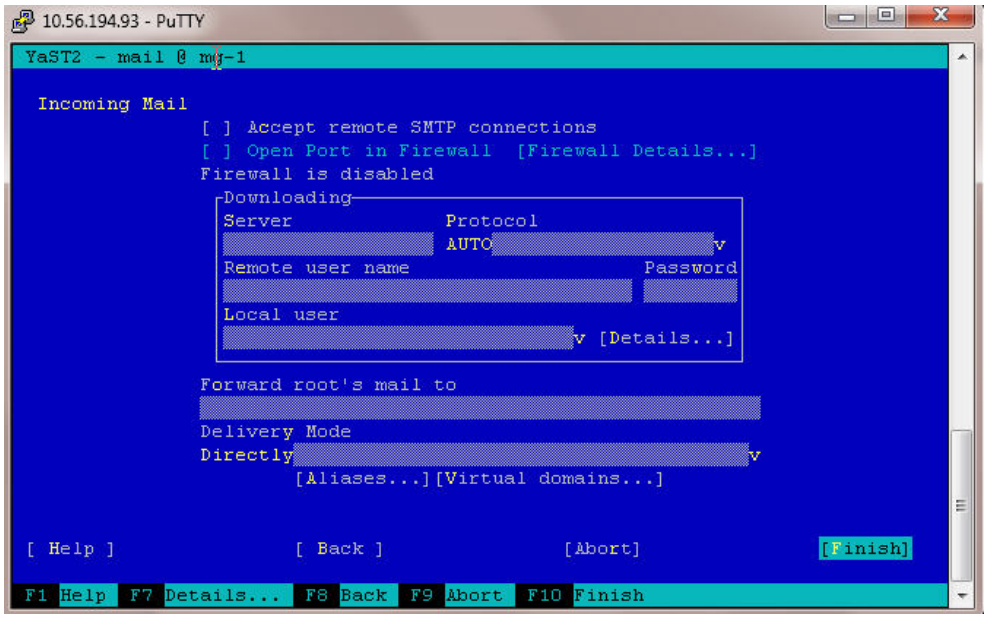
Step 7 The Outgoing Mail window appears. From this window press **Alt-O** to select the Outgoing Mail Server text box and enter the FQDN of the server to use as the email relay server, for example outbound.cisco.com. Press **Alt-N** to move to the next window.

Figure 8-9 Outgoing Mail Window



Step 8 The Incoming Mail window appears. Do not make any changes in this window. Press **Alt-F** or **F10** to finish and save the configuration.

Figure 8-10 Incoming Mail Window



Step 9 From the YaST Control Center window, press **F9** or **Alt-Q** to quit the program.

Step 10 Exit from the SSH window.

- **Export CDR:** The platform tracks each cache-out session and writes out a CDR record that is related to the session when it ends. The CDR recording files are created periodically and can be retrieved using FTP using an 'anonymous' user-id. No password is required, even if prompted. The set of CDR files are created and managed periodically.

  The following is an example of a CDR record:

```
14-05-15
16:00:01.696(ID1),HTTP(ID2),crl.omniroot.com(ID4),/PublicSureServerSV.crl(ID5),
FIN(ID6),1448(ID7),0(ID8),41.188.8.175(ID9),52482(ID10),64.18.20.129(ID11),80(ID12),
286(ID13),15(ID14),FORWARD(ID15),0(ID16),286(ID17),0(ID18),0(ID19),0(ID20),0(ID21),
0(ID22),0(ID23),0(ID24),0(ID25),0(ID26),0(ID27),0(ID28),0(ID29),0(ID30),0(ID31),
0(ID32),0(ID33),0(ID34),0(ID35),0(ID36),0(ID37),0(ID38),0(ID39),0(ID40),0(ID41),
1(ID42),1448(ID43),51314(ID44),(ID45),51DF9719(ID46),49999976(ID47),43384832(ID48),
43(ID49),CDR_EOL (ID50)
```

Note For more information on the CDR records, refer to the *Cisco Videoscape Distribution Suite Transparent Caching Release 5.7.3 CDR Guide*.

The CDR Fields table describes the key CDR parameters.

Table 8-1 CDR Fields

ID	Name	Type	Definition	Length	Nullable	Comments
1	Date Time	Date	This indicates the actual time frame	22	No	Format DD-MM-YY HH24:MI:SS.MS.
2	Protocol	String	This is the format that is used for transmitting data from one device to another.	1-10	No	See the Protocol Values table.
3	Hash	Hex string	The hash is used to generate a unique fixed length data output. This acts as a shortened reference to the original data.	40	Yes	E500FDF95C27ACE 5C86068A6BE603A 0-BA857CBBB
4	Host	String	This is a physical network node that runs both the server and client programs. The host field value from the HTTP Request.	Up to 256	Yes	This only exists on HTTP and Netflix protocols. For example: xml.alexa.com
5	Request Uri	String	The HTTP Request - Uri value from the HTTP start line (web address).	Up to 1K	Yes	For example: /smartart/Text Effects/Texteffect None.swf
6	Session termination reason	String	The “STR” details possible reasons for session termination.	Variable	No	See the Session Termination Reason table in the *Cisco Videoscape Distribution Suite Transparent Caching Release 5.7.3 CDR Guide*.
7	Transferred bytes	Long	The transferred byte amount.		No	
8	Session duration	Integer	The time measure of a session		No	Seconds
9	Source IP	4 octets separated by a dot(IPv4) 16 octets separated by a colon (IPv6)	The source IP is the subscriber IP address.	16	No	IPv4 or IPv6 source IP addresses. For example,110.100.13. 152 or 0001::2000:0000:000 0:0002:01ce
10	Source port	Integer	The source port is the subscriber port.		No	

Table 8-1 CDR Fields (continued)

ID	Name	Type	Definition	Length	Nullable	Comments
11	Destination IP	4 octets separated by a dot (IPv4) 16 octets separated by a colon (IPv6)	The destination IP is the endpoint IP address of the of the host.	16	No	IPv4 or IPv6 destination IP addresses.
12	Destination port	Integer	The destination port is the endpoint of the host.		No	
13	HTTP Service Category ID	Integer	This is the first category index from the matched HTTP service. This is configured within the Policy Manager.		No	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
14	HTTP Service ID	Integer	This is the matched HTTP Service index. This is configured within the Policy Manager.		No	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
15	CDR Type	String	A Cache/Content Data Record field which classifies by type. For more information, see the CDR Types table.		No	The field classifies the CACHE_OUT, VERIFY and VERIDFY_AT.
16	Application Category ID	Integer	The matched Application Category index which is configured within the Policy Manager (Application). The Application field provides a high level summary of the content that can be used for reporting and policy creation, such as file sharing and software updates.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
17	Service Category ID	Integer	The matched Service Category Index which is configured within the Policy Manager.		Yes	The Service Category ID is a legacy configuration based on the policy manager configuration.

Table 8-1 CDR Fields (continued)

ID	Name	Type	Definition	Length	Nullable	Comments
18	Internet Side Network Category ID	Integer	The matched Internet Side Network Category Index which is configured within the Policy Manager. This is associated to a sub-network address.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
19	Subscriber Network Category ID	Integer	The matched Subscriber Network Category Index which is configured within the Policy Manager. The Subscriber Network is a category used to define the destination IP or subnet of the content.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
20	Device Category ID	Integer	The matched Device Category Index which is configured within the Policy Manager. The User Agent is an HTTP header that specifies the type of browser/ device.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
21	HTTP Referrer Category ID (not used, always 0)	Integer	The matched HTTP Referrer Category Index which is configured within the Policy Manager.		Yes	This field is not used. The default value of this field is 0.
22	Content Type (MIME) Category ID	Integer	The matched Content Type (MIME) Category Index which is configured within the Policy Manager.		Yes	This field is not used. The default value of this field is 0.
23	SmartFilter Category ID	Integer	The matched SmartFilter Category Index which is configured within the Policy Manager. This is used as a screen to filter/block specific content to the users.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
24	Video Stream Bit Rate	Integer	This contains the video stream bit.		Yes	
25	Video Stream Resolution	Integer	This contains the video stream resolution.		Yes	For YouTube traffic, the rate value is the itag URL parameter value that indicates the video resolution.

Table 8-1 CDR Fields (continued)

ID	Name	Type	Definition	Length	Nullable	Comments
26	Bandwidth Policy ID	Integer	This is a bandwidth policy index. It populates when a Rule with a bandwidth policy is matched.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
27	Admission Policy ID	Integer	This is an Admission policy index. It populates when a Rule with an Admission policy is matched. This is associated with blocking or re-directing sites.			The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
28	Cache Control Policy ID	Integer	This is a Cache Control policy index. It populates when a Rule with a Cache Control policy is matched. This policy disables/enables the media cache traffic to be cached in Cisco VDS TC Manager.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
29	Webcache Control Policy ID	Integer	This is a web cache control policy index. It populates when a rule with a web cache control policy is matched. This policy disables the web cache traffic to be cached in Cisco VDS TC Manager.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
30	DSCP Policy ID	Integer	This is a DSCP policy index. It populates when a Rule with a DSCP policy is matched. This policy classifies cache out traffic in DSCP field that exists in the IPv4/IP6 headers.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
31	Bandwidth Policy Rule ID	Integer	This is the matched rule index that applies the bandwidth policy on the current session.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.

Table 8-1 CDR Fields (continued)

ID	Name	Type	Definition	Length	Nullable	Comments
32	Admission Policy Rule ID	Integer	This is the matched rule index that applies the admission policy on the current session.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
33	Cache Control Policy Rule ID	Integer	This is the matched rule index which applies the cache control (media cache control) policy on the current session.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
34	Web-Cache Control Policy Rule ID	Integer	This is the matched rule index which applies the webcache control policy on the current session.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
35	DSCP Policy Rule ID	Integer	This is the matched rule index which applies the DSCP policy on the current session.		Yes	The values can be seen in VDS TC Manager. This field is used by VDS TC Analytics for Traffic Analysis.
36	Cache-In - (Number of Requests)	Integer	This field is used to specify the number of requests cached from in the traffic.		Yes	This is used in Web-Cache Sessions Only.
37	Cache-In (Number of Bytes)	Integer	This field is used to specify the number of bytes cached from in the traffic.		Yes	This is used in Web-Cache Sessions Only.
38	Cache-In (Duration)	Integer	This field is used to measure the amount of expired time of the incoming cache from the traffic.		Yes	This is used in Web-Cache Sessions Only.
39	Cache-Out (Number of Requests)	Integer	This field is used to specify the number of cached -out requests from the traffic.		Yes	This is used in Web-Cache Sessions Only.
40	Cache-Out (Number of Bytes)	Integer	This field is used to specify the number of cached -out bytes from the traffic.		Yes	This is used in Web-Cache Sessions Only.
41	Cache-Out (Duration)	Integer	This field is used to measure the amount of expired time of the cache -out from the traffic.		Yes	This is used in Web-Cache Sessions Only.

Table 8-1 CDR Fields (continued)

ID	Name	Type	Definition	Length	Nullable	Comments
42	Forward (Number of Requests)	Integer	The process of forwarding requests directly to the requesting subscriber without caching it.		Yes	This is used in Web-Cache Sessions Only.
43	Forward (Number of Bytes)	Integer	The process of forwarding the number of bytes directly to the requesting subscriber without caching it.		Yes	This is used in Web-Cache Sessions Only.
44	Forward (Duration)	Integer	The time expended during the forwarding process.		Yes	This is used in Web-Cache Sessions Only.
45	Referrer	String	This is an HTTP header field that identifies the address of the web page linked to the resource being requested.		Yes	This is used in Web-Cache Sessions Only.
46	HTTP Agent Hash	Hex string	The HTTP agent hash is used to generate the fixed length output data of the HTTP agent that acts as a shortened reference to the original data.			See HTTP Agent Hash section for more information.
47	Full File Size	Integer	Displays available user requested file size in bytes.			
48	Cached File Size	Integer	Displays the amount of bytes cached from the requested user file.			
49	Connection_ignore_reason	Integer	Displays the reason a connection state changed			
50	CDR EOL	String	This is the end of the line for the CDR.		Yes	

- **VDS TC uses iSCSI and NetApp E2724 or IBM DS3524:** The VDS TC platform is fully integrated with the NetApp E2724 or IBM DS3524 storage solution. The NetApp E2724 can consolidate up to sixteen (16) fully redundant hosts and expand to support up to 13.2 TB of data on (using 600 GB SAS drives). The IBM DS3524 can consolidate up to sixteen (16) fully redundant hosts and expand to 5.7 TB of data (using 380 GB SAS drives). Both the NetApp and IBM SAN solution provide a wizard-based installation, intuitive management, and advanced data protection software.

  Supporting the NetApp E2724 or IBM DS3524 increases the storage capacity of VDS TC platform and allows the creation of a clustered file-system (using two or more VDS TC cache engines).

Traffic Specific Features

Using Type of service (ToS) values you can color packets to enable shapers and other network devices to use these "color" indicators to determine how to handle traffic. For example you may want to color certain types of traffic to prevent that traffic from being shaped by a downstream device.

The following is an example of how to color only CACHE_OUT traffic with a ToS value of 8. Place this configuration in the <service> section of the configuration file:

```
<tos_markup>
    <tos_overwrite>1</tos_overwrite>
    <class name="cache-out">
        <out_mode>CACHE_OUT</out_mode>
    </class>
    <action name="mark-with-8">
        <dscp>8</dscp>
    </action>
    <rule id="1">
        <enable_rule>1</enable_rule>
        <priority>20</priority>
        <class_name>cache-out</class_name>
        <action_name> mark-with-8</action_name>
    </rule>
</tos_markup>
```

This example enables ToS coloring with the <tos_overwrite>1</tos_overwrite> field. The coloring rule that is created in this example (rule id="1") will set the ToS bits to 8 for traffic that matches the "cache-out" class. Also the rule is given a priority of 20. If more than one rule exists, the rules are executed in order, from the lowest priority number to the highest priority number.

Fine Tuning System Behavior for Mobile Operators

The VDS TC platform has a built-in mechanism that is able to adjust to the network behavior changes because of network delays and network congestions. You must configure all of the following parameters in the <tcp> subsection of the <service> section in the configuration file to not lose packets on transmission. An explanation for each parameter is provided in Table 8-2.

```
<tcp>
<conn_idle_timeout>60</conn_idle_timeout>
<tcp_snd_buf>393216</tcp_snd_buf>
<min_cwnd_in_packets>3</min_cwnd_in_packets>
<min_ssthresh_in_packets>15</min_ssthresh_in_packets>
<slow_start_reduce_ratio>7</slow_start_reduce_ratio>
<http_timeout_sec>60</http_timeout_sec>
</tcp
```

Table 8-2 TCP parameters

Parameter	Definition
<conn_idle_timeout>*seconds***</conn_idle_timeout>**	The seconds after which the connection is removed from the internal connection table, as a result of no activity on that session. In the example this is 60 seconds.
<tcp_snd_buf>*no_bytes***</tcp_snd_buf>**	The maximum number of bytes that the caching application sends (in cache-out) before receiving any acknowledgment from the client (in flight bytes). In the example this is 393,216.
<min_cwnd_in_packets>*no_pkts***</min_cwnd_in_packets>**	The minimum number of packets to which the TCP congested window can drop. This parameter affects only the cache-out sessions. The higher the number, the more packets that can be sent even if the client did not receive them or is congested. In the example this is 3.
<min_ssthresh_in_packets>*no_pkts***</min_ssthresh_in_packets>**	The minimum number of packets to which the TCP slow start boundary can drop. This parameter affects only cache-out sessions. The higher the number, the faster a TCP connection can recover after packet loss. The slow_start_reduce_ratio parameter determine the factor that the caching application should reduce. In the example this is 15.
<slow_start_reduce_ratio>*factor***</slow_start_reduce_ratio>**	This configures the factor that the caching application can use to reduce the TCP slow start threshold upon packet loss (in cache-out mode only). The number is a factor of ten percent. For example, 5 would set the ratio to 50%, which is the default TCP behavior. The maximum value is ten. The higher the number, the faster a TCP connection recovers from a packet loss. In the example this is 7, which would set the factor to 70%.
<http_timeout_sec>*seconds***</http_timeout_sec>**	This value defines after how many seconds of no activity the session is removed from the internal session table. The default is 20 seconds.

Table 8-2 TCP parameters

Parameter	Definition
<use_new_stack_params>1**</use_new_stack_params>**	This parameter is not currently supported.
<scaling_window_multiplier>*multiplier***</scaling_window_multiplier>**	This parameter is not currently supported.

Controlling Core Dumps

A caching application core dump is when the caching application writes everything from its memory to disk. This can consume a lot of disk space, especially as the memory for the caching application is increased.

The default behavior is for the caching application to perform a core dump. To configure the caching application to not perform a core dump, add the following text to the configuration file in the <policy> subsection of the <service> section:

```
<execute_debug_info>0</execute_debug_info>
```

Setting the value to 0 disables the core dump. To re-enable the core dump feature, set the value to **1**.

Configurations Using the CLI

The following sections describe commands that you can use from the VDS TC CLI to configure the VDS TC device.

This section includes the following information:

- Managing Passwords
- Recovering Passwords
- Configuring the Management Network
- Configuring Time
- Managing the Caching Service
- Managing Servers
- Resetting the Management Service
- Managing Traffic Detection

Caution The commands and their options are case sensitive.

Managing Passwords

To set or change the passwords to access regular mode and enable mode, use the following commands:

Command	Purpose
console# **access user-password** *new_password*	Establishes a new password or changes an existing password for the regular command level, also referred to as the user password.
console# **access enable-password** *new_password*	Establishes a new password or changes an existing password for the enable mode.

Recovering Passwords

If you forget the initial CLI password or the enable mode password, you can reset them to their default values using the special Rescue CLI. The Rescue CLI is available only from the serial console. For more information, *see* CLI Modes, page 7-3.

Configuring the Management Network

To be able to access the CLI using SSH and a LAN connection, you must configure the IP address and default gateway address for the management server. You can configure an IPv4 address, an IPv6 address, or both. In enable mode, follow these steps to configure this information

Table 8-3 Configuring a Management IPv4 Address:

	Command	Purpose
Step 1	console# **config**	Enters Configuration mode.
Step 2	configuration# **network ip** *IP_address netmask*	Configures the management IPv4 address of the VDS TC device. You must enter the subnet mask using the dotted decimal notation.
Step 3	configuration# **network default_gw** gateway_address	Configures the default IPv4 gateway address for the VDS TC to use.
Step 4	configuration# **exit**	Returns to enable mode.
Step 5	configuration# **apply**	Applies the configurative changes.
Step 6	console# **ping** [**-c** *count*] [-I Source_*interface_or_sourceIP_address*] *destination_IP*	Tests network connectivity. After you configure the network settings, you should test the configuration. This command is available in both regular and enable mode. **Note** If you do not specify the number of times to repeat the ping with the count option, the ping will continue until you press **Ctrl-C**.
Step 7	console# **traceroute** *destination_IP*	Test network connectivity and displays the router hops that packets will actually take when traveling to their destination. This command can help troubleshoot network connectivity and is a good troubleshooting step if the ping fails.

Table 8-4 Configuring a Management IPv6 Address

	Command	Purpose
Step 1	console# **config**	Enters Configuration mode.
Step 2	configuration# **network ip6** *ipv6_address/ipv6_prefix*	Configures the management IPv6 address of the VDS TC device.
Step 3	configuration# **network default6_gw** *gateway_address*	Configures the default IPv6 gateway address for the VDS TC device to use.
Step 4	configuration# **apply**	Applies the configurative changes.
Step 5	configuration# **exit**	Returns to enable mode.
Step 6	console# **ping** [**-c** *count*] [-I Source_*interface_or_sourceIP_address*] *destination_IP*	Tests network connectivity. After you configure the network settings, you should test the configuration. This command is available in both regular and enable mode. **Note** If you do not specify the number of times to repeat the ping with the count option, the ping will continue until you press **Ctrl-C**.
Step 7	console# **traceroute** *destination_IP*	Test network connectivity and displays the router hops that packets will actually take when traveling to their destination. This command can help troubleshoot network connectivity and is a good troubleshooting step if the ping fails.

Configuring Time

NTP is a networking protocol that is used to synchronize clocks. You can use NTP to configure computer systems with the IP address of an NTP time source server.

You can manually configure the local time on VDS TC management server or you can configure the VDS TC management server to point to an NTP server.

Manually Configuring Time

To manually configure the time on the VDS TC management server, use the following command in Configuration mode:

Command	Purpose
configuration# **time** *MMDDYYhhmm*	Sets the system date and time. Two numbers are used to represent each part: month, day, year, hour, and minutes. For example **time 1104120815** would set the date to November 4, 2012 8:15 am.

Note The CLI configuration# prompt requires Enable mode privileges. For more information, *see* Chapter 9, "Using CLI Commands (Cluster)".

To verify that the time that is currently set on the VDS TC device, enter **show time** in either regular mode or enable mode.

Using an NTP Server

To configure the VDS TC management server to use an NTP server, refer to the "Configuring an NTP server and time zone" in the Platform Operational Specific Features section.

Managing the Caching Service

From the CLI you can stop or start the caching service. To manage the caching service use the **oper service** command in enable mode.

oper service {**stop** | **start**}

Syntax Description

stop	Stops the caching service.
start	Starts the caching service after it has been stopped.

Managing Servers

From the CLI you can stop, start, or restart a single server.

Follow these steps to stop, start, or restart a single server:

Step 1 From enable mode, enter **oper server** *server_number*, where *server_number* is the number of the server that you want to manage. After you enter this command, the oper server *X#* prompt will appear, where *X* is the number of the server that you entered. For example, oper server 1#.

Step 2 Enter one of the following commands depending on what you want to do with the server:

- **stop:** Stops the server's operation.
- **start:** Starts the server after it has been stopped.
- **restart:** Performs a soft reload of the caching service by stopping and restarting. Use this option to restart the caching service on the specific cache engine with minimal impact to the transit traffic.

Resetting the Management Service

In Configuration mode, only one user at a time can perform a configuration operation.

If your active terminal session does not respond because another user is already performing a configuration operation, you can reset the management service to recover the ability to configure the system.

Caution Use this option with caution. If you reset the management service, all open CLI sessions are reset, including the session from which you are executing the command, and any configurations that were not applied will be lost.

To reset the management service use the following command in enable mode:

Command	Purpose
console# **reset**	Resets all active CLI sessions in the system.

Managing Traffic Detection

From the CLI you can manage traffic detection by categorizing traffic types. *See* Working with the Configuration Files, page 7-5.

File-Based Configuration

When configuring the software settings in the main configuration file, you can configure fields in the <common> section to apply to all servers, or in the specific section for an individual server. When there is a value for a field in both sections, the value of the field in the individual server section takes priority over the value of the field in the common section for that server.

After making changes to the configuration files, you must apply your changes. See Working with the Configuration Files, page 7-5.

Note Any changes that you make to the VDS TC configuration file will only take affect after you import the configuration file into the VDS TC management server. For information on how to import this file, see Working with the Configuration Files in Chapter 7, "Working with Cisco VDS TC Management Tools (Cluster)".

This section includes the following configuration information:

- Configuring SNMP
- Configuring P2P Protocols
- Configuring Bandwidth Management
- Traffic Forwarding Modes
- Configuring Caching Policies
- Configuring Virtual IP Address
- NIC Flapping Option
- Applying Configuration Changes
- Upgrading the VDS TC Software
- Updating the VDS TC License

Configuring SNMP

To configure the SNMP settings add or edit the following fields in the <common> section of the configuration file:

```
<snmp>
```

```
    <trap-ip>IP_Address_SNMP_Server</trap-ip>
    <snmp-read-community>read_community_string</snmp-read-community>
    <snmp-write-community>write_community_string</snmp-write-community>
    <snmp-trap-community>trap_community_string</snmp-trap-community>
</snmp>
```

For example the configuration might look like the following:

```
<snmp>
<trap-ip>10.11.12.1</trap-ip>
<snmp-read-community>gdcbhv</snmp-read-community>
<snmp-write-community>nkppui</snmp-write-community>
<snmp-trap-community>ffff</snmp-trap-community>
</snmp>
```

Configuring P2P Protocols

To enable or disable the different P2P protocol support, add or edit the following fields in the <common> section or <blade id=x> section, in the <service> <protocols> subsection of the configuration file:

- BitTorrent

  <enable-bittorent>*value***</enable-bittorent>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.

- BitTorrent uTP

  <enable-utp-bittorent>*value***</enable-utp-bittorent>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.

Note Currently uTorrent 3.4.2 is supported.

- eDonkey

  <enable-edk>*value***</enable-edk>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.

- Ares

  <enable-ares>*value***</enable-ares>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.

- HTTP

  <enable-http>*value***</enable-http>**

  Replace *value* with **1** to enable the protocol or **0** to disable the protocol.

For example, to enable ares support, add the following to the <common> section or <blade id=x> section of the configuration file:

```
<enable-ares>1</enable-ares>
```

Configuring Bandwidth Management

To configure bandwidth management, add or edit the following fields in the <common> section or <blade id=x> section, <service> subsection of the configuration file:

```
<bandwidth-management>
    <enable-bandwidth-management>value<enable-bandwidth-management>
    <bandwidth-per-connection>max_bw_per_IP<bandwidth-per-connection>
</bandwidth-management>
```

Replace *value* with **1** to enable the bandwidth management service or **0** to disable the bandwidth management service. Replace *max_bw_per_IP* with the maximum bandwidth per connection, in bytes/second, that you want to support.

Traffic Forwarding Modes

The traffic forwarding mode controls whether the VDS TC system will forward traffic in promiscuous mode or bounce mode:

- **Promiscuous:** The L2/L3 switch forwards traffic via two dedicated ports without changing L2 addresses (as-is).
- **Bounce:** The platform sends packets back using the same interface while swapping the source and destination MAC addresses.

To configure this setting, edit the following field in the <common> or <blades>1 section, <service> <net> subsection of the configuration file:

```
<fwd-mode>mode</fwd-mode>
```

Replace *mode* with either **BOUNCING** or **PROMISC**.

Configuring Caching Policies

The caching policy indicates the percentage of upstream P2P traffic that must come from the internal cache. To configure this setting, add or edit the following fields in the <service> section of the configuration file:

```
<policy>
    <upload_cache_out>%_of_traffic</upload_cache_out>
</policy>
```

Replace %_of_traffic with one of the following:

- **0:** Disables this feature. All upstream traffic can come from local peers.
- **1-99:** This is the specified percentage of the upstream traffic that must come from the VDS TC cache storage and the remainder of the upstream traffic can come from local peers.
- **100:** Upstream traffic can only come from the internal cache.

Configuring Virtual IP Address

Each individual server has multiple virtual IP addresses that are used by the L3 switch for health monitoring, load sharing, and next hop addresses. The L3 switches query the virtual IP addresses with ICMP requests to verify the health of each interface and the overall server availability.

You can configure one virtual IP address per interface, per server which enables you to load distribute redirected traffic between the different interfaces of the caching engine. Each server can have up to ten virtual IP addresses. The virtual IP addresses that you configure on the VDS TC host must match the IP addresses on the L3 switch.

To configure the virtual IP address of an interface, add or edit the following field under the <blade id=x> section, <cache-engine> <network><network_interfaces> <nic nic_index=> subsection of the interface: interface:

```
vip>IP_address</vip>
```

For example:

```
<nic nic_index="0">
    <name>eth4</name>
    <nic_detail>IFF_PF_PACKET</nic_detail>
    <vip>10.138.201.1</vip>
</nic>
```

NIC Flapping Option

You can configure the VDS TC system to initiate flapping whenever the system starts. To configure the NIC flapping option, add or edit the following in <service> section of the configuration file:

```
<interface_flapping>
    <set_sleep_interfaces>1</set_sleep_interfaces>
    <shut_down_iff_delay>delay_1</shut_down_iff_delay>
    <start_up_iff_delay>delay_2</start_up_iff_delay>
</interface_flapping>
```

Enter a value in seconds for *delay_1* and *delay_2* to configure a shut_down_iff delay value and a start_up_iff delay value, respectively. The default value for the shut_down_iff delay is 10 seconds and the default value for the start_up_iff delay is 5 seconds.

The the shut_down_iff delay and start_up_iff delay values are used as follows when you configure the NIC flapping option:

Step 1 When the system starts, the system waits for a period of time equal to the shut_down_iff delay value and then it performs an ifdown for all interfaces that are currently up.

Step 2 The system then waits for a period of time equal to the start_up_iff delay value and then it does an ifup to all interfaces that were up in Step 1.

Applying Configuration Changes

After changing the configuration file as required, you can activate the new configuration.

During the activation process, VDS TC verifies the new configuration, and flags any errors found in the log.

Follow these steps to activate the new configuration file:

Step 1 Use VDS TC Manager to upload the new configuration file to VDS TC. *See* the *Cisco Videoscape Distribution Suite Transparent Caching Manager User Guide* (part number OL-28017-02) for more details.

Step 2 Open an SSH session to the VDS TC CLI and log in using the username **admin** and password.

Step 3 At the CLI prompt, enter **enable** to enter Enable mode.

Step 4 At the CLI prompt, enter **config** to access Configuration mode.

Step 5 At the CLI prompt, enter **import localhost** *filename* where *filename* is the name of the new configuration file that you uploaded in Step 1.

Note After performing this step, the new configuration is downloaded to the system, but it is not yet applied. The new configuration is applied only after you complete all of the activation steps and the system is restarted.

Step 6 The following options are available for displaying, applying, or discarding the new configuration, or for restoring an old configuration:

Command	Purpose
`console# show config`	Displays the currently loaded configuration.
`configuration# display`	Displays the new configuration that was loaded, but not yet applied.
`configuration# diff`	Displays the differences between the current configuration and the new configuration.
`configuration# apply`	Applies the new configuration in place of the current configuration.
`configuration# discard`	Discards the new configuration without making any changes to the current configuration.
`configuration# restore`	Restores the old configuration after applying a new configuration.

Step 7 When the **apply** command reports a restart service exit the configuration mode using the **exit** command.

Note The new configuration is applied only after all the activation steps are completed, and the system is restarted.

Step 8 Log off of the VDS TC CLI by entering the **exit** command twice.

The configuration session is complete.

Also *see* Working with the Configuration Files, page 7-5.

Upgrading the VDS TC Software

For information on upgrading the VDS TC application, see the *Cisco Videoscape Distribution Suite Transparent Caching Application Upgrade Guide* at http://www.cisco.com/c/dam/en/us/td/docs/video/videoscape/distribution_suite/vds/v5_7_3/VDS-TC_5.7.3_app_upgrade_guide.pdf.

Updating the VDS TC License

To view information about your license from the CLI, in Enable mode enter **show license**. Information about the installed license is displayed, including the version number and enabled features.

Follow these steps to install a new software license:

Caution If you are also upgrading the VDS TC software, follow the steps in the Cisco Videoscape Distribution Suite Transparent Caching Application Upgrade Guide for updating the VDS TC License. These steps are to update a license to add additional features or limits.

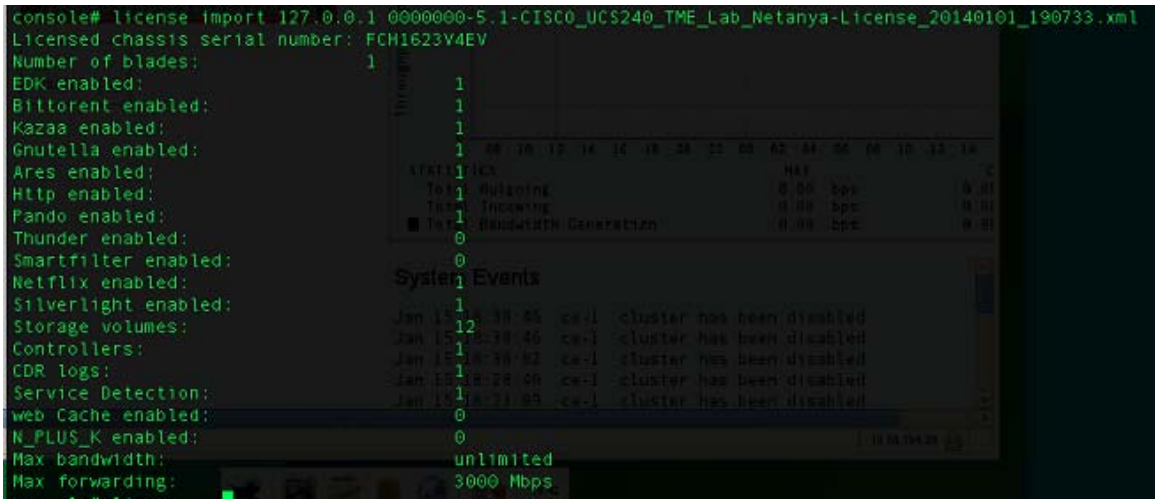
Step 1 Copy the new license file to your TFTP server.

Step 2 From the VDS TC prompt, enter the **enable** command. When prompted, enter the Enable mode password and press **Enter**.

Step 3 From Enable mode, enter the command **license import** *127.0.0.1 filename*, where *filename* is the name of the new license file, for example 0000000-5.7-CISCO_UCS240_XY-License_20140127_19862.xml.

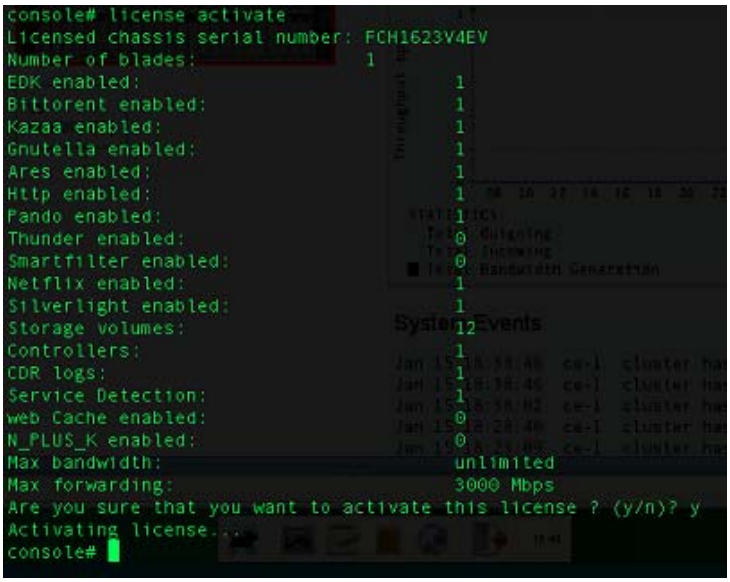
Note You must be in Enable mode, *not* Configuration mode to import a new license.

Figure 8-11 License Import



Step 4 Enter **license activate** to apply the license. When prompted with "Are you sure you want to activate this license?" enter **Y**.

Figure 8-12 License Activation



Step 5 Use VDS TC Manager to confirm the upgrade and version number. In a web browser, enter the management IP address of the VDS TC appliance to connect to the VDS TC Manager.

Step 6 Enter a username of **padmin** and the password that was provided by Cisco.

Step 7 Choose **Configuration > License Manager** to confirm the license has been upgraded.

Configuring TACACS+ on the Server

TACACS+ is a an access control network protocol that allows user authentication and authorization with the customer's TACACS+ server for both the VDS TC Manager environment and the VDS TC CLI (using Telnet or SSH).

To be able to work with TACACS+ authentication and authorization, you must configure several parameters on the TACACS+ server and in the VDS TC management environment.

Note If the configured TACACS+ server is not available, or if a TACACS+ server is not configured, authentication and authorization is performed using the standard VDS TC management server username and password built-in mechanism.

Configuring TACACS+ for VDS TC Support, Using Cisco Secure ACS Release 4.x

To configure TACAC+ support for the VDS TC platform on the Cisco Secure ACS server, perform the following steps on the Cisco Secure ACS server:

Step 1 Create a network device group for the VDS TC systems:

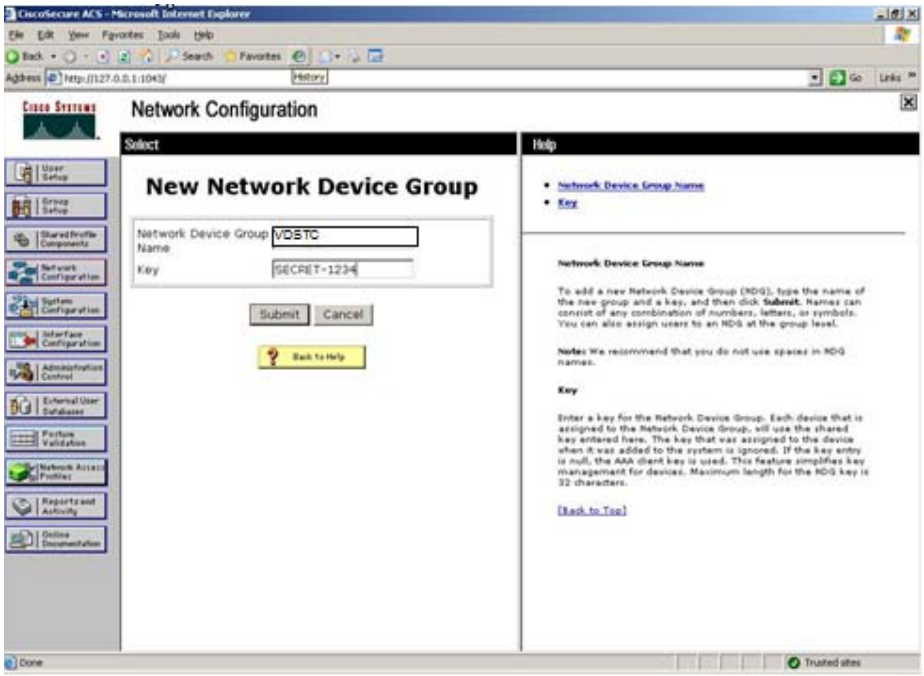
a. Click **Network Configuration**.

b. Click **Add Entry** to add a new network device group.

c. Enter a name for the network device group and in the Key field enter the shared key.

d. Click **Submit**.

Figure 8-13 Cisco Secure ACS New Network Device Group



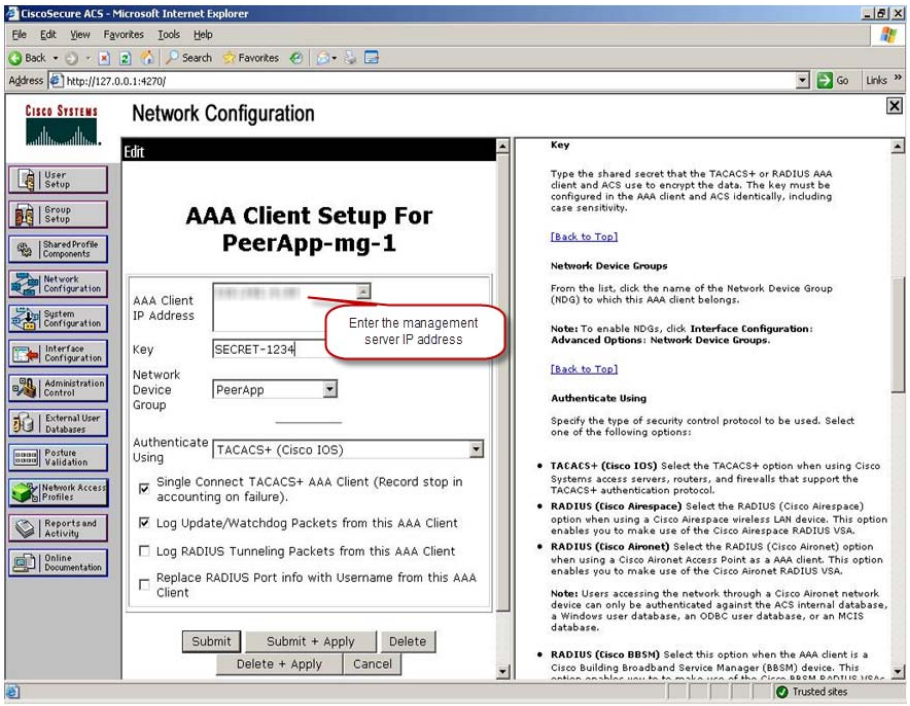
Step 2 For each VDS TC system that will use TACAC+, create a AAA client, including the management IP address of the VDS TC system and a shared secret (password):

a. Click **Network Configuration**.

b. Click the network device group you created in Step 1 and click **Add Entry**.

c. In the Add AAA Sever window, enter the IP address of the VDS TC system, enter the shared secret key, and choose **TACACS+(Cisco IOS)** from the Authenticate Using drop-down list.

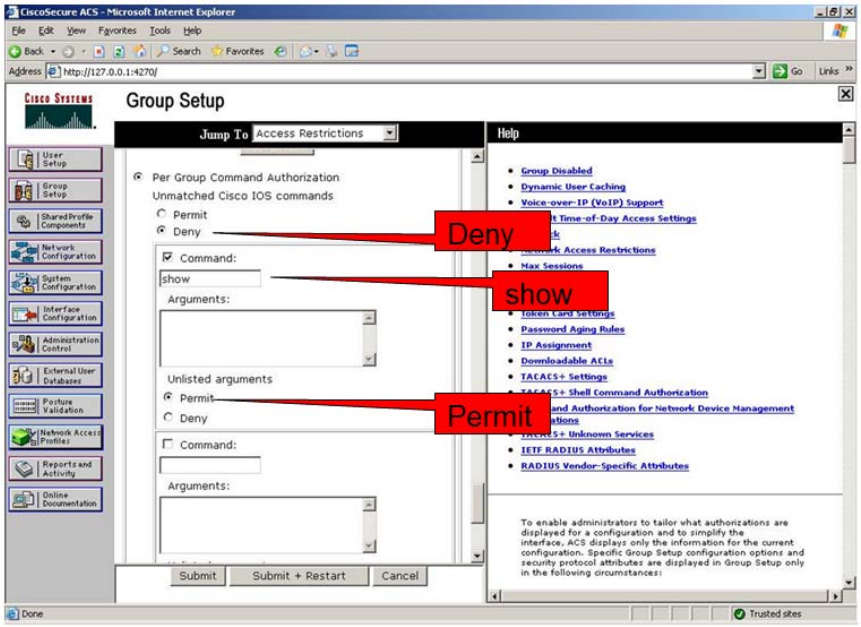
Figure 8-14 Cisco Secure ACS AAA Client Setup



Step 3 Configure authorization for the users. Each user in the TACACS+ server that should have access to VDS TC must be assigned to one of the following groups. You must create these groups on the TACACS+ server, with the following parameters.

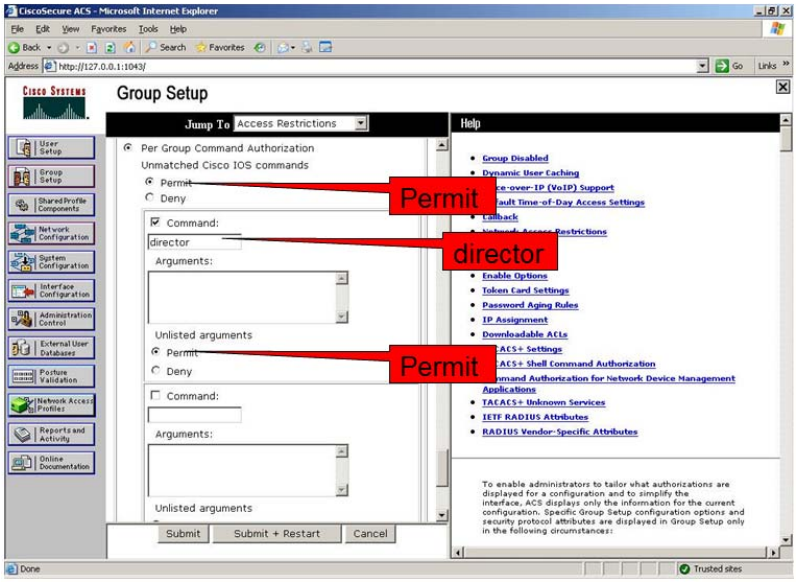
- **VDSTC-standard:** A user that is associated with this group can be used for standard VDS TC CLI login and standard VDS TC Manager login. For the CLI, it allows the user access to non privileged commands. It also allows the user access to the standard VDS TC Manager user interface. Configure the following settings for this group.
  - Group name: **VDSTC-standard**
  - Set the Per Group Command Authorization for this group to **Deny**.
  - Check the **Command** check box for this group and enter **show** in the command text box.
  - For the Unlisted Arguments setting, click **Permit**.

Figure 8-15 Cisco Secure ACS Group Setup: VDSTC-standard



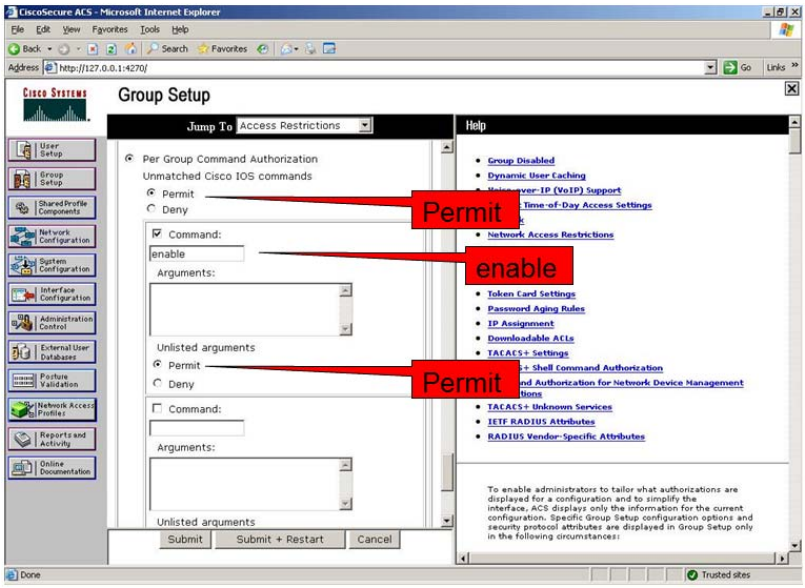
- **VDSTC-director:** A user that is associated with this group can modify the content of the caching application policies, managed inside the VDS TC Manager. Users that are not a member of this group will have view-only privileges for the caching application policy environment and will not be able to modify the policies. Configure the following settings for this group.
  - Group name: **VDSTC-director**
  - Set the Per Group Command Authorization for this group to **Permit**.
  - Check the **Command** check box for this group and enter **director** in the command text box.
  - For the Unlisted Arguments setting, click **Permit**.

Figure 8-16 Cisco Secure ACS Group Setup: VDSTC-director



- **VDSTC-privileged:** A user that is associated with this group can be used for Enable mode CLI command access and VDS TC Manager login. Configure the following settings for this group.
  - Group name: **VDSTC-privileged**
  - Set the Per Group Command Authorization for this group to **Permit**.
  - Check the **Command** check box for this group and enter **enable** in the command text box.
  - For the Unlisted Arguments setting, click **Permit**.

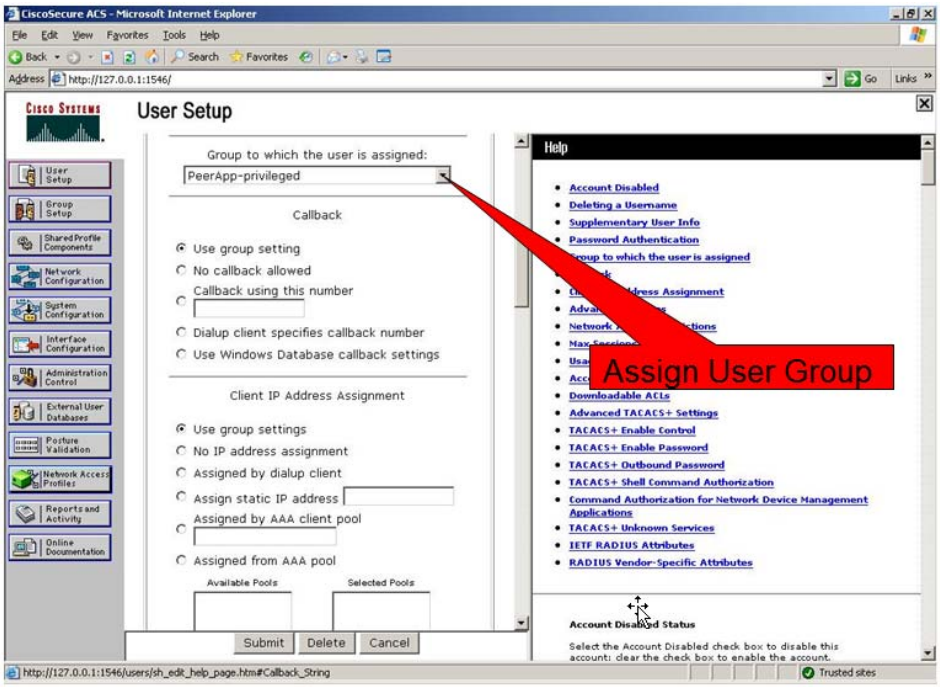
Figure 8-17 Cisco Secure ACS Group Setup: VDSTC-privileged



Step 4 Configure the VDS TC users on the TACACS+ server and configure the following settings:

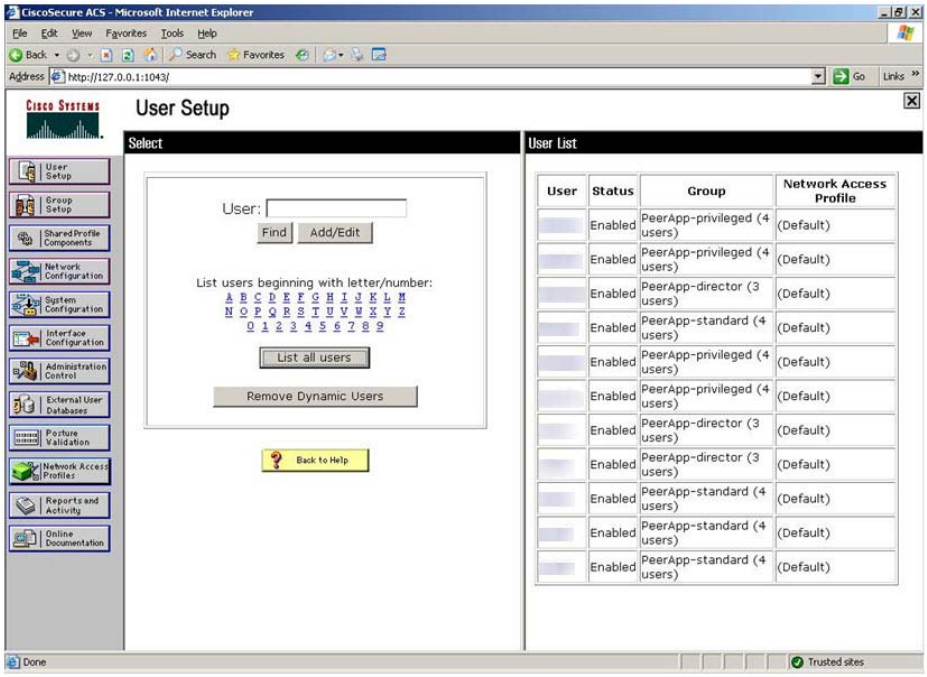
- In the User Setup section in the CiscoSecure PAP area, enter a password in the Password and Confirm Password text boxes. Currently only clear passwords are supported.
- Choose a group to assign to the user from the Group to Which User is Assigned drop-down list box. Assign the user to either the VDSTC-standard, VDSTC-director, or VDSTC-privileged group depending on the privileges they should have.
- In the Advanced TACACS+ Settings section choose **Use Cisco secure PAP Password**.

Figure 8-18 User Setup



Step 5 Once users are associated with the correct privileges, you can view the groups the users are assigned to from the User List, as shown in the following figure:

Figure 8-19 Cisco Secure ACS User List

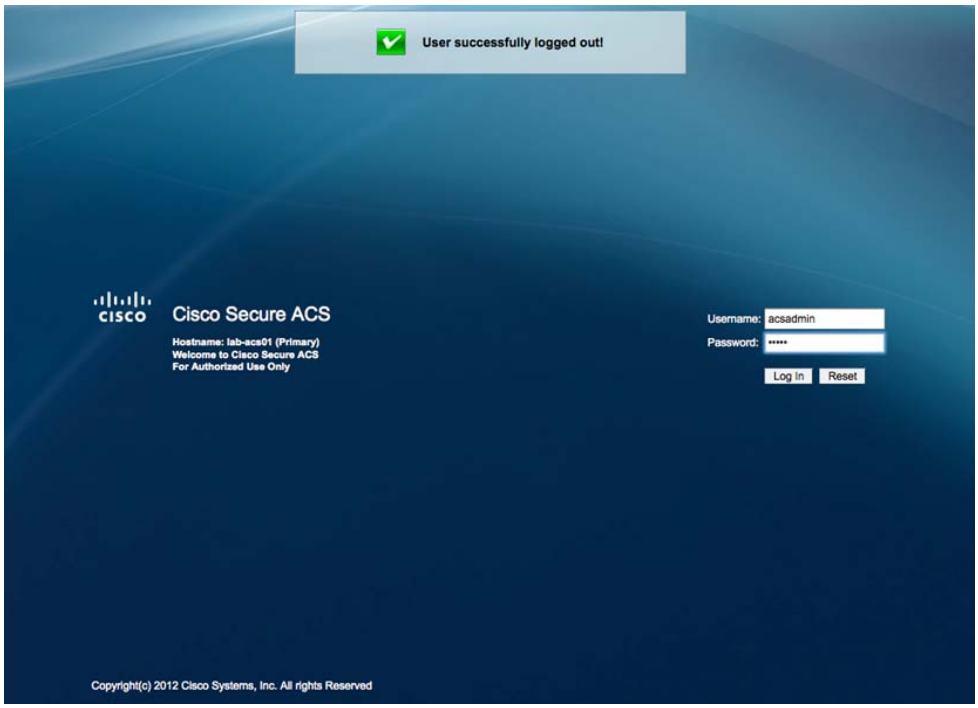


Configuring TACACS+ for VDS TC Support, Using Cisco Secure ACS Release 5.4

To configure TACAC+ support for the VDS TC platform on the Cisco Secure ACS server, perform the following steps on the Cisco Secure ACS server:

Step 1 Using HTTPS, log into the Cisco Secure ACS server. When prompted, enter your username and password for the server.

Figure 8-20 Cisco Secure 5.4 ACS Login Window



Step 2 From the Cisco Secure ACS main window, choose **Network Resources > Network Device Groups > Network Devices and AAA Clients**.

Step 3 Click Create and in the window that appears, enter the following information:

a. **Name:** Enter the name of the VDS TC system.

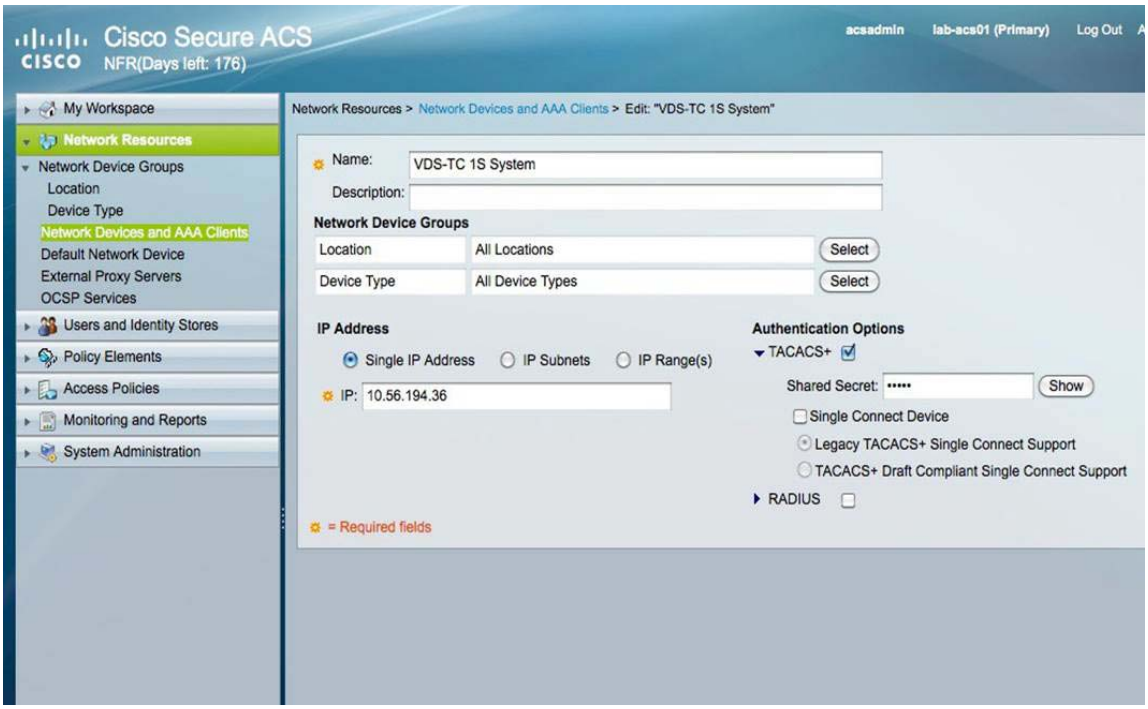
b. Authentication options:

- Check the **TACACS+** check box.
- In the Shared Secret field, enter the shared secret that corresponds to the shared secret that you will configure in the <tacacs_secret> tag in the cluster configuration file. See the Configuring the VDS-TC Management Server for TACACS+ for configuring the cluster configuration file for this value.

c. Click the **Single IP Address** radio button and in the IP address field, enter the IP address of the VDS TC management server.

Step 4 Click **Submit**. Your results should look similar to the following figure.

Figure 8-21 Create New AAA Client



Step 5 Next you will create three identity groups, one for Admins, one for directors, and one for the help desk team. To create these groups, choose **Users and Identity Stores > Identity Groups**.

Step 6 Click **Create**, enter the following values for the first group:

a. **Name:** TC Admins

b. **Parent:** All Groups (this is the default)

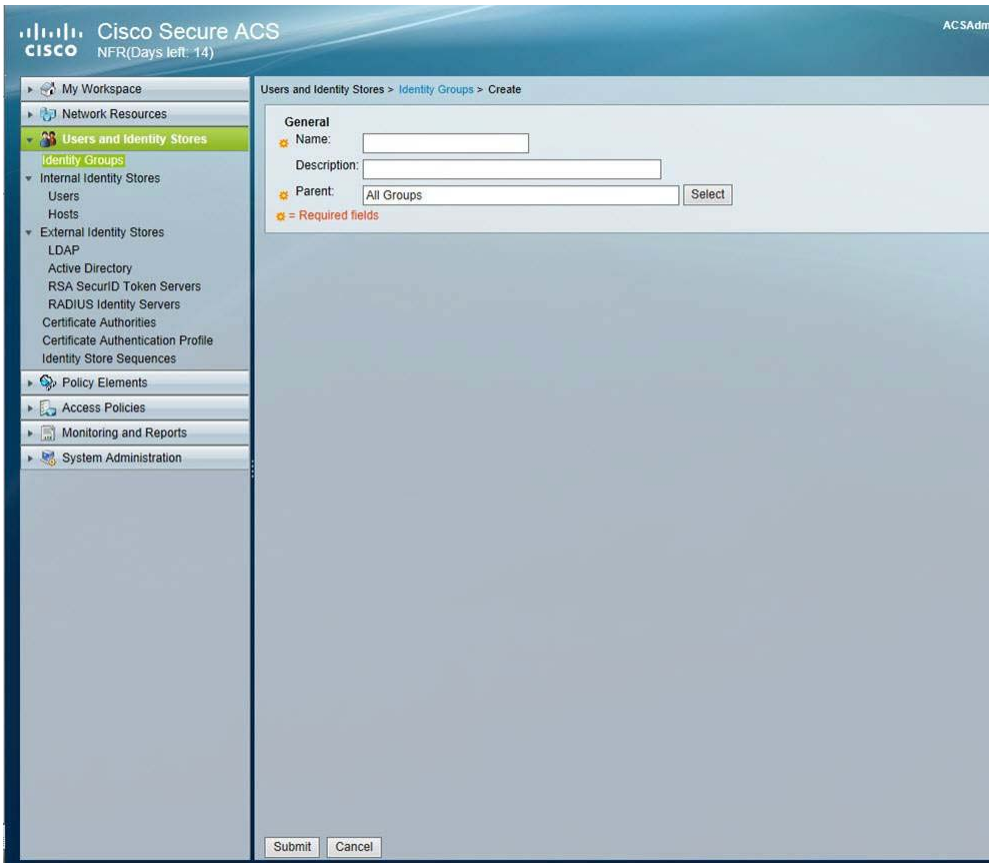
c. Click **Submit**.

Step 7 Repeat Step 6 to create the following groups:

- TC Director
- TC Help Desk

Note The group names can be any names that you would like to use, but the following configuration steps will use TC Admin, TC Director, and TC Help Desk as examples.

Figure 8-22 Create Identity Groups



Step 8 Next you will create three users, one for Admin access, one for Director access, and one for Help Desk access. To create these users, choose **Users and Identity Stores > Internal Identity Stores > Users**.

Step 9 Click **Create** to create the Admin user and complete the following information:

a. **Name:** Enter the name for the Admin user, for example tcadmin.

b. **Identity Group:** Choose the Identity Group that you created for the Admin group, which was All Groups: TC Admins in our example.

c. Click **Submit**.

Step 10 Click Create to create the Director user account:

a. **Name:** Enter the name for the Director user, for example tcdirector.

b. **Identity Group:** Choose the Identity Group that you created for the Director group, which was All Groups: TC Director in our example.

c. Click **Submit**.

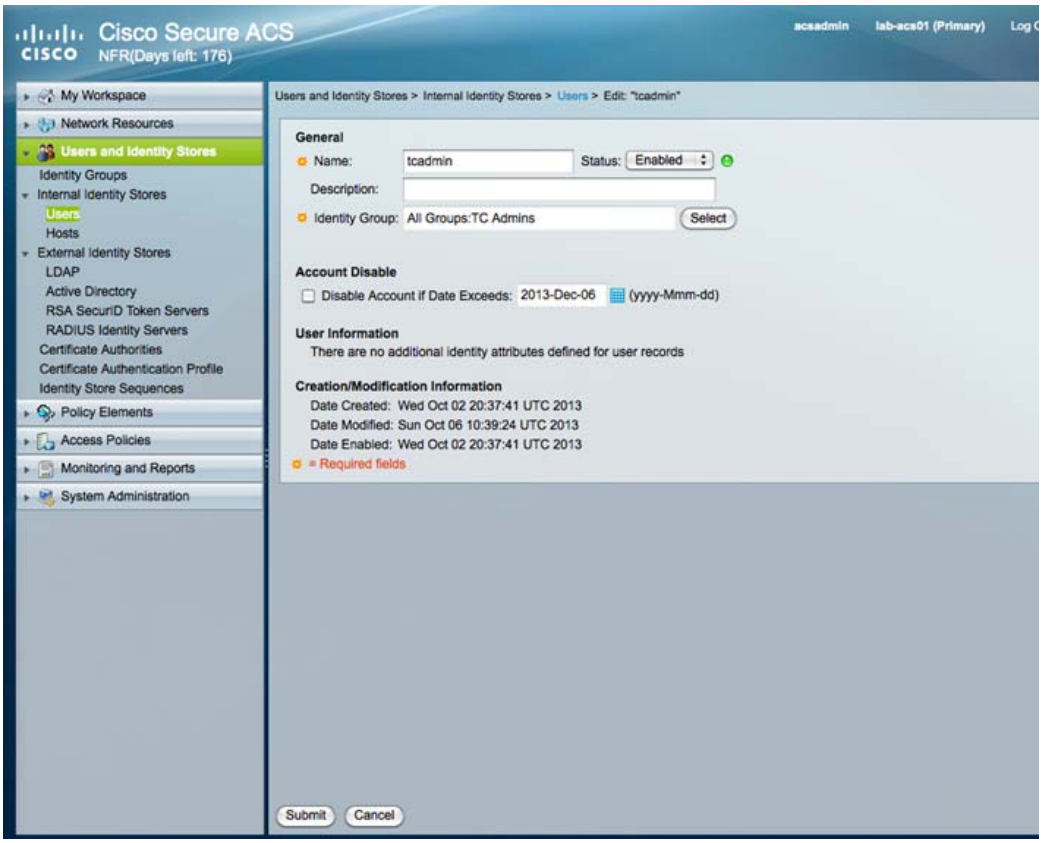
Step 11 Click Create to create the Help Desk user account:

a. **Name:** Enter the name for the Help Desk user, for example tchelp.

b. **Identity Group:** Choose the Identity Group that you created for the Help Desk group, which was All Groups: TC Help Desk in our example.

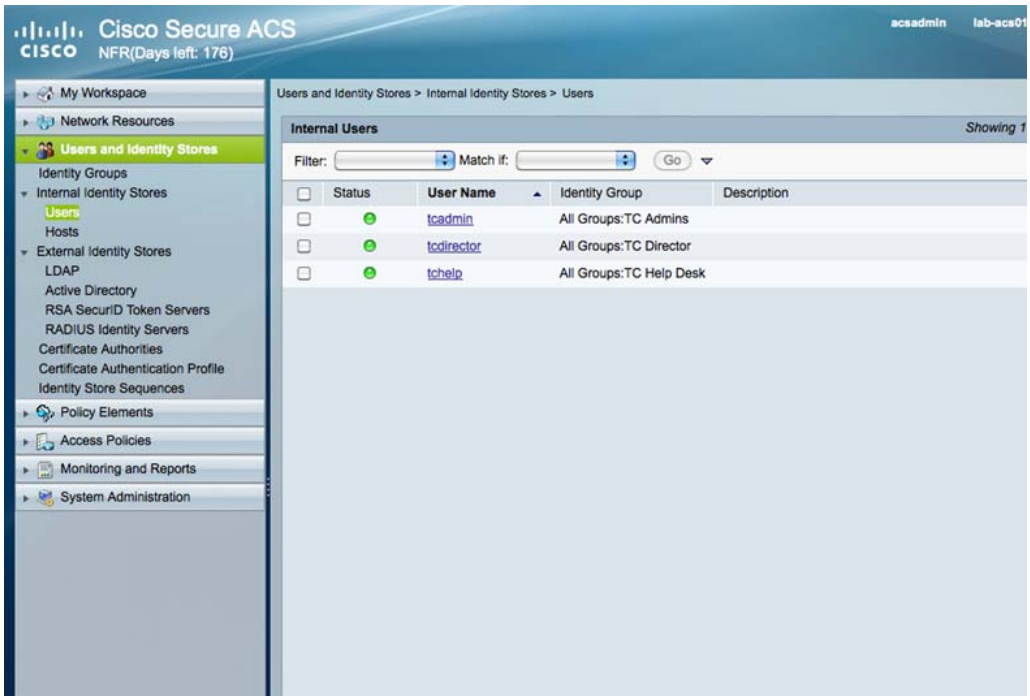
c. Click **Submit**.

Figure 8-23 Create User



Step 12 After you are done creating the users and groups, the output should look similar to the following:

Figure 8-24 Users List



Configure Policy Elements

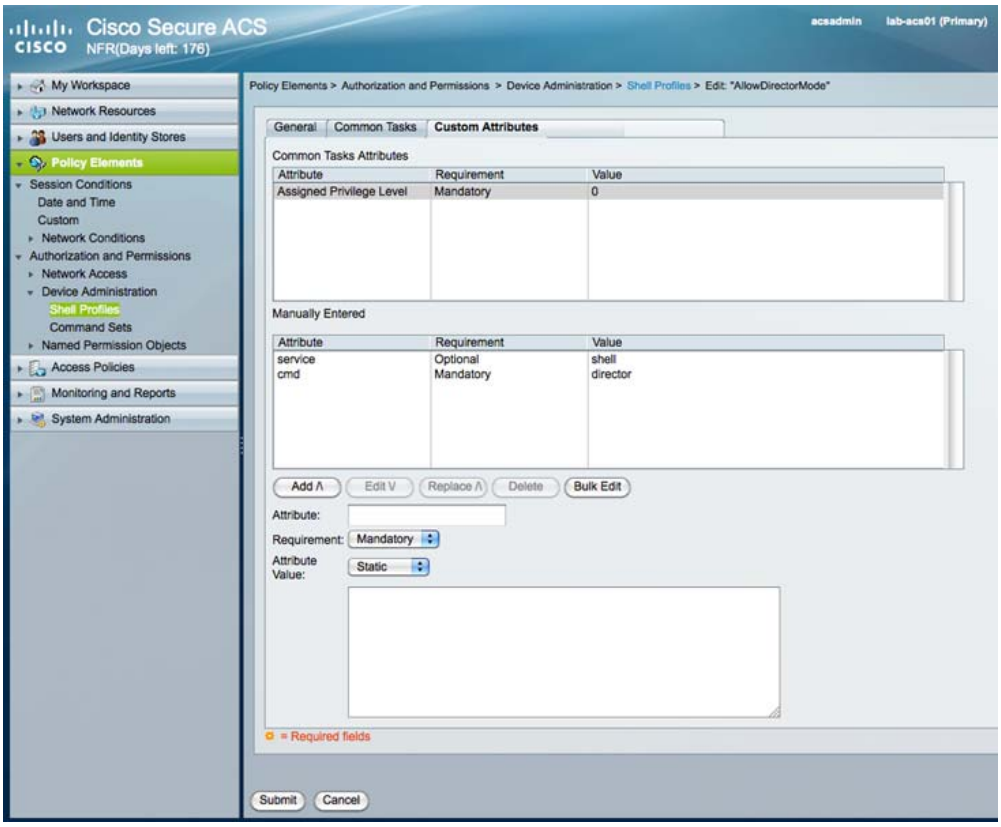
Follow these steps to create the policy elements in Cisco Secure ACS 5.4:

Step 1 Choose **Policy Elements > Authorizations and Permissions > Device Administration > Shell profiles**.

Step 2 Click **Create** and enter the following values to create a Director shell profile:

- On the General tab, in the Name field enter **AllowDirectorMode**.
- On the Common Tasks tab, configure the following:
  - Default Privilege: **Static** with a value of **0**
  - Maximum Privilege: **Not in use**
  - All Shell Attributes should be set to **Not in Use**.
- On the Custom Attributes tab add the following attributes:
  - attribute: service, requirement: optional, value: shell
  - attribute: cmd, requirement: mandatory, value: director

Figure 8-25 Director Custom Attributes



Step 3 Click **Submit**.

Step 4 Choose **Policy Elements > Authorizations and Permissions > Device Administration > Command Sets**.

Step 5 Click **Create** and enter the following values to create the AllowDirectorMode command sets:

- In the Name field, enter **AllowDirectorMode**.
- Make sure the "Permit any command that is not in the table below" check box is *unchecked*.
- Add the director command
  - Choose **Permit**.
  - In Command field enter **director**.
  - Leave Arguments field empty.
  - Click **Add**.
- Add the enable command
  - Choose **Permit**.
  - In Command field enter **enable**.
  - Leave Arguments field empty.
  - Click **Add**.
- Click **Submit**.

Step 6 Click **Create** and enter the following values to create the AllowEnableMode command sets:

- In the Name field, enter **AllowEnableMode**.
- Make sure the "Permit any command that is not in the table below" check box is *unchecked*.
- Add the enable command
  - Choose **Permit**.
  - In Command field enter **enable**.
  - Leave Arguments field empty.
  - Click **Add**.
- Click **Submit**.

Step 7 Click **Create** and enter the following values to create the DenyEnableMode command sets:

- In the Name field, enter **DenyEnableMode**.
- Make sure the "Permit any command that is not in the table below" check box is *unchecked*.
- Add the enable command
  - Choose **Permit**.
  - In Command field enter **show**.
  - Leave Arguments field empty.
  - Click **Add**.
- Click **Submit**.

Configure Access Policies

Follow these steps to configure the access polices:

Step 1 Choose **Access Policies > Access Services > Default Device Admin**.

Step 2 Under Policy Structure, check the **Identity** and **Authorization** check boxes.

Step 3 Click **Submit**.

Step 4 From the Allowed Protocols tab, check only the **Allow PAP/ASCII** check box.

Step 5 Click **Submit**.

Step 6 Choose **Access Policies > Access Services > Default Device Admin > Identity**.

Step 7 On the window that appears, click the **Single Result Selection** radio button.

Step 8 In the Advanced Options section, make sure the following settings are configured:

- If authentication failed: **Reject**
- If user not found: **Reject**
- If process failed: **Drop**

Step 9 If you made any changes to the Access Policies > Access Services > Default Device Admin > Identity window, click **Save Changes**.

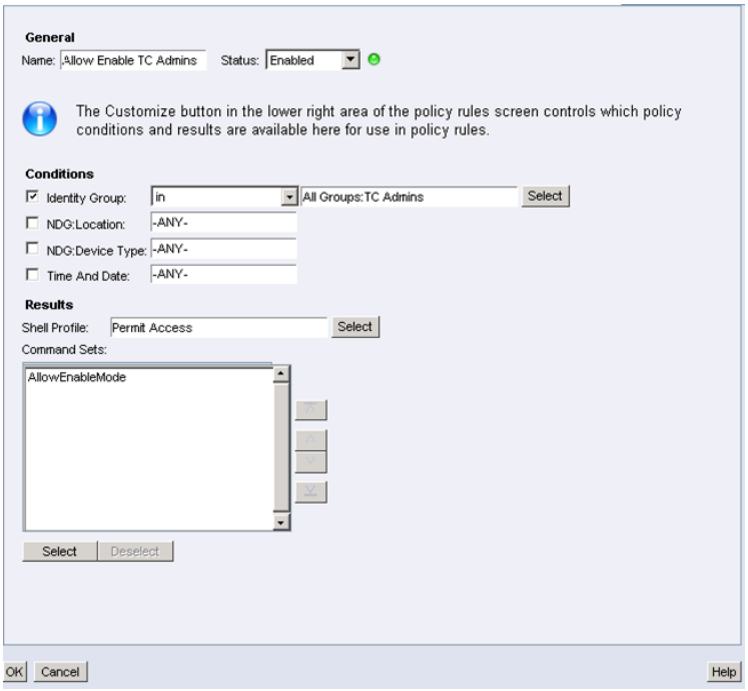
Step 10 Choose **Access Policies > Access Services > Default Device Admin > Authorization**.

Step 11 Click **Create** to create a rule. In the window that appears, enter the following values:

- Name: **Allow Enable TC Admins**
- Status: **Enabled**

- Identity Groups: **Check** the check box, choose **in** from the drop-down list and click **Select** and choose the group that you created for the Admins group. In our example, this is TC Admins.
- Shell Profile: Select **Permit Access**
- Commands Sets: Select **AllowEnableMode**
- Click **OK**.

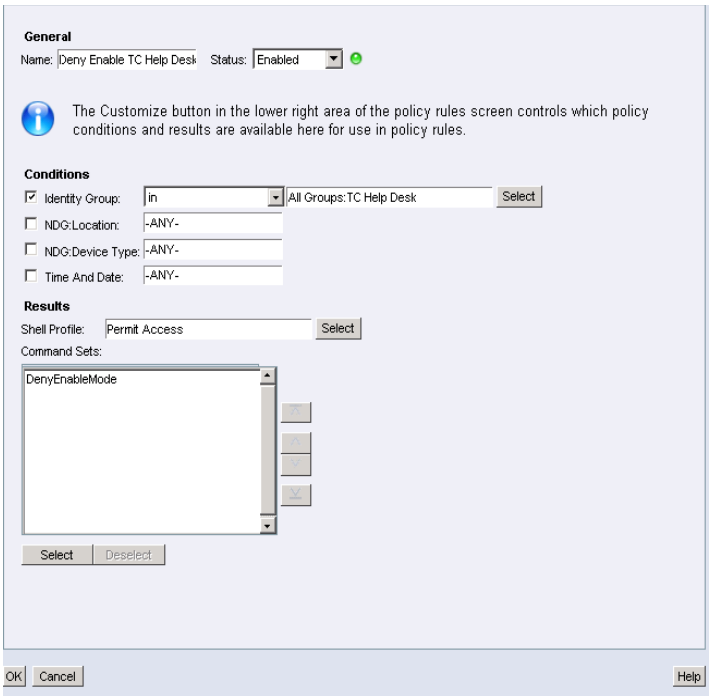
Figure 8-26 Allow Enable TC Admins Settings



Step 12 Click **Create** to create a rule. In the window that appears, enter the following values:

- Name: **Deny Enable TC Help Desk**
- Status: **Enabled**
- Identity Groups: **Check** the check box, choose **in** from the drop-down list and click **Select** and choose the group that you created for the Help Desk group. In our example, this is TC Help Desk.
- Shell Profile: Select **Permit Access**
- Commands Sets: Select **DenyEnableMode**
- Click **OK**.

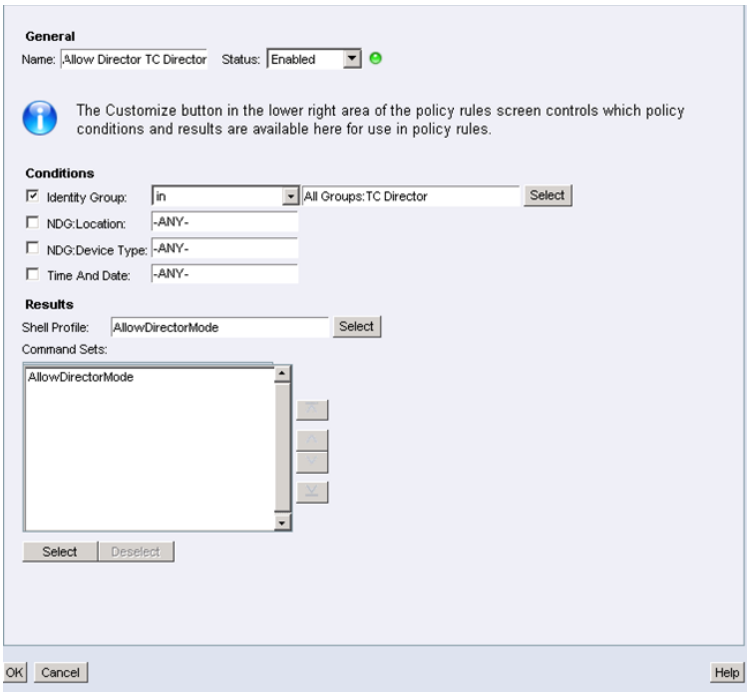
Figure 8-27 Deny Enable TC Help Desk Settings



Step 13 Click **Create** to create a rule. In the window that appears, enter the following values:

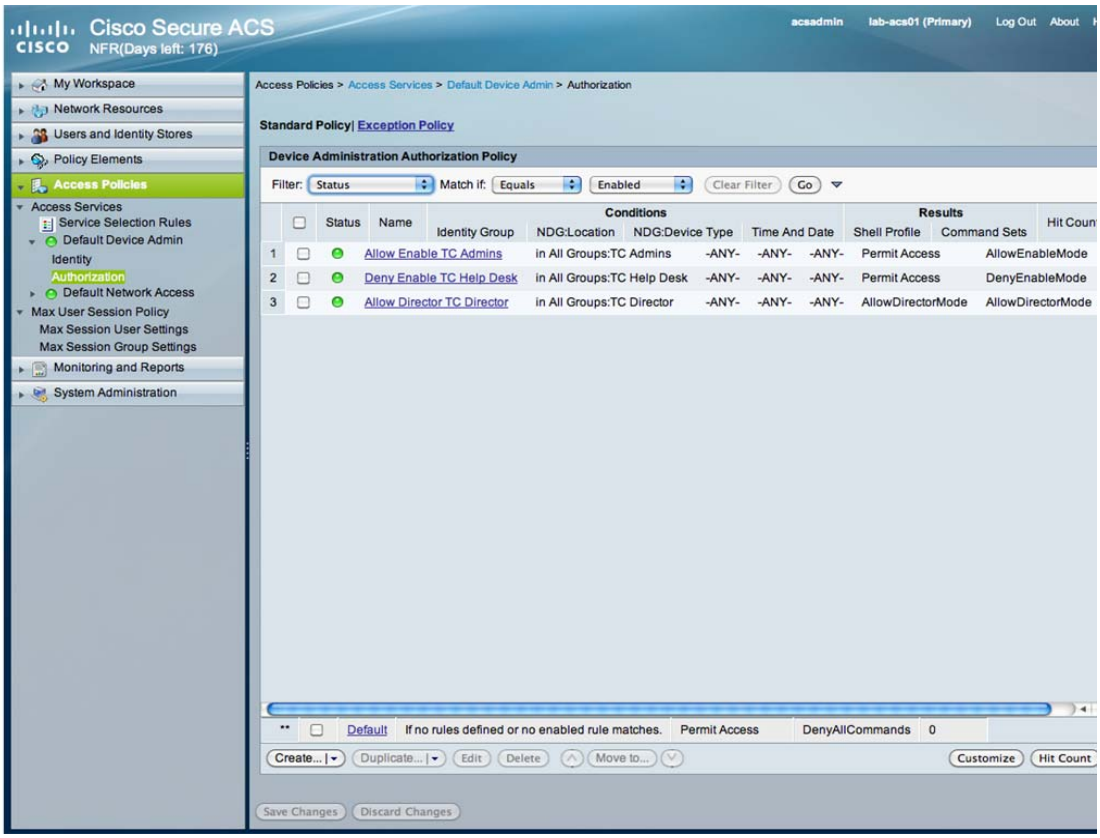
- Name: **Allow Director TC Director**
- Status: **Enabled**
- Identity Groups: **Check** the check box, choose **in** from the drop-down list and click **Select** and choose the group that you created for the Director group. In our example, this is TC Director.
- Shell Profile: Select **AllowDirectorMode**
- Commands Sets: Select **AllowDirectorMode**
- Click **OK**.

Figure 8-28 Allow Director TC Director Settings



Step 14 When you are done creating the authorization polices, the window should look like the following:

Figure 8-29 Authorization Settings



Configuring the VDS-TC Management Server for TACACS+

Configure and license the system (*see* the *Cisco Videoscape Distribution Suite Transparent Caching Installation Guide* [part number OL-28015-02]) and add the following configuration to the <mgmt-config> section in the VDS TC configuration file, where *server_ip* is the IP address of the TACACS+ server and *secret* is the shared secret that is configured on the TACACS+ server for the VDS TC platform:

```
<TACACS_configuration>
    <tacacs_server_ip>server_ip</tacacs_server_ip>
    <tacacs_secret>secret</tacacs_secret>
</TACACS_configuration>
```

For example:

```
<TACACS_configuration>
    <tacacs_server_ip>10.1.1.65</tacacs_server_ip>
    <tacacs_secret>SECRET-1234</tacacs_secret>
</TACACS_configuration>
```

Note The TACACS+ secret is a string that is configured on both sides and must match on the TACACS+ Server and the VDS TC management server.

After the TACACS+ configuration is applied, TACACS+ users can log into the VDS TC CLI using Telnet or SSH, or log into the VDS TC Manager.

Note Any changes that you make to the VDS TC configuration file will only take affect after you import the configuration file into the VDS TC management server. For information on how to import this file, see Working with the Configuration Files in Chapter 7, "Working with Cisco VDS TC Management Tools (Cluster)"

Using CLI Commands (Cluster)

The VDS TC platform is controlled using a set of CLI commands, allowing full control over its operational modes. The CLI commands are divided into three categories:

- **Regular mode commands:** From this mode you can display version and licensing information for the VDS TC platform, and you can access and manipulate the system log. You cannot perform system configuration changes from this mode.
- **Enable mode commands:** From this mode you have full control over the system configuration, cache content manipulation, networking behavior, licensing, platform operation state, and you can manage the software version on which the system runs.
- **Configuration mode commands:** From this mode you can make platform configuration changes. Changes you make are stored but are implemented only when you use the **apply** command.

To access the VDS TC CLI use the username **admin.** The default password is platform specific.

The following is a sample authentication session:

```
Login as: admin
Using keyboard-interactive authentication.
Password:
Cli version - 5.7.3b54
Snmp version - VDS-TC Transparent Caching mgmt software version 5.7.3b54
console>
```

Critical user activities are recorded in the system eventlog, allowing administrators to monitor system activity.

This chapter provides a reference for all of the CLI commands that are available for an Integrated Appliance solution. See CLI Reference for a full list of available CLI commands.

Regular Mode Commands

This section describes the commands that are available in Regular mode. These commands are also available in Enable mode. The Regular Mode Commands table lists the commands that are described in this section.

Note All of the CLI commands work with both IPv4 and IPv6.

Note All commands and their parameters are case sensitive.

Table 9-1 Regular Mode Commands

Command	Description
arp	Displays the ARP table
current_cli_users	Displays the list of VDS TC admin users who are currently logged into the VDS TC CLI
direction	Calculates the visible subnets on the interface
dmesg	Displays the message buffer
enable	Enters Enable mode
eventlog	Provides access to event log operations
exit	Exits the current mode
help	Displays the list of available commands for the mode you are in when you execute the command
ifconfig	Displays the interface(s)
iostat	Displays extended I/O statistics
jumbo	Sends jumbo echo messages
ping	Sends echo messages
show	Displays run-time information
traceroute	Displays the route used by the packet to reach its destination

arp

To display the ARP table, use the **arp** command.

arp

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **arp** command:

```
console> arp
Address                  HWtype  HWaddress           Flags Mask            Iface
ce-16                    ether   d4:8c:b5:4d:a9:32   C                     bond0.50
10.56.194.1              ether   00:00:0c:07:ac:c8   C                     bond0
10.11.18.204             ether   00:80:e5:36:0e:60   C                     bond0.60
10.11.18.206             ether   00:80:e5:2f:ed:4e   C                     bond0.60
ce-15                    ether   a4:4c:11:29:b8:9c   C                     bond0.50
10.11.18.208             ether   00:80:e5:36:40:54   C                     bond0.60
```

```
10.56.194.99            ether   3c:08:f6:60:71:92   C                    bond0
ce-1                    ether   a4:93:4c:aa:46:e8   C                    bond0.50
ce-8                    ether   d4:8c:b5:4d:a7:30   C                    bond0.50
ce-7                    ether   a4:93:4c:aa:8e:f6   C                    bond0.50
10.11.18.205            ether   00:80:e5:36:10:a6   C                    bond0.60
10.11.18.203            ether   00:80:e5:2f:f8:ae   C                    bond0.60
10.11.18.207            ether   00:80:e5:36:10:c0   C                    bond0.60
10.11.18.200            ether   00:80:e5:2f:d8:3c   C                    bond0.60
ce-4                    ether   d4:8c:b5:4d:a7:4c   C                    bond0.50
ce-2                    ether   d4:8c:b5:4d:de:04   C                    bond0.50
10.11.18.209            ether   00:80:e5:36:47:42   C                    bond0.60
ce-3                    ether   c4:64:13:39:a8:ec   C                    bond0.50
10.56.194.117           ether   3c:08:f6:60:8b:b6   C                    bond0
ce-9                    ether   d4:8c:b5:4d:98:80   C                    bond0.50
ce-11                   ether   a4:93:4c:aa:6b:8a   C                    bond0.50
ce-14                   ether   fc:99:47:49:c4:ee   C                    bond0.50
10.11.18.201            ether   00:80:e5:36:02:b2   C                    bond0.60
ce-5                    ether   d4:8c:b5:4d:a5:1e   C                    bond0.50
ce-12                   ether   d4:8c:b5:4d:cf:4e   C                    bond0.50
ce-13                   ether   50:57:a8:e1:a1:96   C                    bond0.50
10.11.18.202            ether   00:80:e5:2f:f5:5c   C                    bond0.60
ce-10                   ether   a4:4c:11:2a:0c:c0   C                    bond0.50
ce-6                    ether   d4:8c:b5:4d:b7:d8   C                    bond0.50
```

current_cli_users

To display the admin users who are currently logged on, use the **current_cli_users** command.

current_cli_users

Note You must be logged on using an admin username and password to view the output from this command. This command only displays admin users that are logged directly into the VDS TC CLI. It does not display users that have used the sudo command from the Linux CLI to access the VDS TC CLI.

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following example displays a list of users currently logged on to the system:

```
console> current_cli_users
admin    pts/1        Apr  4 21:32 (10.21.150.101)
```

direction

To calculate the visible subnets on the specified interface, use the **direction** command.

direction *interface_name*

Syntax Description

interface_name	The interface for which you want to display the subnets, for example eth0.

Command Modes

Regular mode and Enable mode

Examples

The following sample displays the visible subnets on the interface eth0 using the **direction** command:

```
console> direction eth0
tcpdump: WARNING: eth0: no IPv4 address assigned
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
1000 packets captured
1000 packets received by filter
0 packets dropped by kernel
    913 10.11.12.0/24
     16 10.11.18.0/24
      2 10.11.80.0/24
     68 10.56.10.0/24
      1 64.101.47.0/24
```

dmesg

To displays the message buffer of the kernel, use the **dmesg** command.

dmesg

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **dmesg** command:

```
console> dmesg
usb 3-1.1: Manufacturer: ATEN International Co. Ltd
usb 3-1.1: configuration #1 chosen from 1 choice
input: ATEN International Co. Ltd CS1716A V1.0.098 as /class/input/input9
input: USB HID v1.00 Keyboard [ATEN International Co. Ltd CS1716A V1.0.098] on
usb-0000:00:1d.1-1.1
input: ATEN International Co. Ltd CS1716A V1.0.098 as /class/input/input10
input: USB HID v1.00 Device [ATEN International Co. Ltd CS1716A V1.0.098] on
usb-0000:00:1d.1-1.1
input: ATEN International Co. Ltd CS1716A V1.0.098 as /class/input/input11
input: USB HID v1.10 Mouse [ATEN International Co. Ltd CS1716A V1.0.098] on
usb-0000:00:1d.1-1.1
usb 1-5.2: USB disconnect, address 6
usb 1-5.1: new high speed USB device using ehci_hcd and address 8
usb 1-5.1: new device found, idVendor=0781, idProduct=5406
usb 1-5.1: new device strings: Mfr=1, Product=2, SerialNumber=3
usb 1-5.1: Product: U3 Cruzer Micro
usb 1-5.1: Manufacturer: SanDisk
```

```
usb 1-5.1: SerialNumber: 40549102FB103472
usb 1-5.1: configuration #1 chosen from 1 choice
scsi3 : SCSI emulation for USB Mass Storage devices
usb-storage: device found at 8
usb-storage: waiting for device to settle before scanning
scsi 3:0:0:0: Direct-Access     SanDisk  U3 Cruzer Micro  8.02 PQ: 0 ANSI: 0 CCS
sd 3:0:0:0: Attached scsi removable disk sdc
sd 3:0:0:0: Attached scsi generic sg3 type 0
usb-storage: device scan complete
usb 1-5.1: USB disconnect, address 8
usb 1-4: new high speed USB device using ehci_hcd and address 9
usb 1-4: new device found, idVendor=0781, idProduct=5406
usb 1-4: new device strings: Mfr=1, Product=2, SerialNumber=3
usb 1-4: Product: U3 Cruzer Micro
usb 1-4: Manufacturer: SanDisk
usb 1-4: SerialNumber: 40549102FB103472
usb 1-4: configuration #1 chosen from 1 choice
scsi4 : SCSI emulation for USB Mass Storage devices
usb-storage: device found at 9
usb-storage: waiting for device to settle before scanning
scsi 4:0:0:0: Direct-Access     SanDisk  U3 Cruzer Micro  8.02 PQ: 0 ANSI: 0 CCS
sd 4:0:0:0: Attached scsi removable disk sdc
sd 4:0:0:0: Attached scsi generic sg3 type 0
usb-storage: device scan complete
SCSI device sdc: 31301631 512-byte hdwr sectors (16026 MB)
sdc: Write Protect is off
sdc: Mode Sense: 45 00 00 08
sdc: assuming drive cache: write through
SCSI device sdc: 31301631 512-byte hdwr sectors (16026 MB)
sdc: Write Protect is off
sdc: Mode Sense: 45 00 00 08
sdc: assuming drive cache: write through
 sdc: sdc1
device eth0 entered promiscuous mode
audit(1250604820.505:2): dev=eth0 prom=256 old_prom=0 auid=4294967295
device eth0 left promiscuous mode
audit(1250604820.517:3): dev=eth0 prom=0 old_prom=256 auid=4294967295
device eth0 entered promiscuous mode
audit(1250604820.537:4): dev=eth0 prom=256 old_prom=0 auid=4294967295
device eth0 left promiscuous mode
audit(1250604820.813:5): dev=eth0 prom=0 old_prom=256 auid=4294967295
```

enable

Enable mode allows you to access CLI commands and make configuration changes. These changes include cache content manipulation, networking behavior, licensing, and managing the software version on which the system runs. To enter enable mode, use the **enable** command.

enable

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode

Examples

The following example shows the use of the **enable** command. After the user enters the correct password, the system enters Enable mode, as indicated by the pound sign (#):

```
console> enable
Password:
console#
```

eventlog

The event log lists all of the log messages sent to the system log by all of the VDS TC service components (applications, CLI and SNMP). To view or export the content of the event log, use the **eventlog** command.

eventlog {**date** *date*| **export** *tftp_server filename* | **show** | **tail**}

Note Additional parameters for this command are available in Enable mode. See eventlog, page 9-29 for a description of the Enable mode parameters.

Syntax Description

date *date*	Displays the eventlog for the date specified with the *date* parameter. The *date* parameter is in the format of DD-MM-YYYY.
export *tftp_server filename*	Exports the content of the event log to the TFTP server with the name or IP address specified with the *tftp_server* parameter.
show	Displays the event log entries.
tail	Displays the online event log entries.

Command Modes

Regular mode and Enable mode

Examples

The following example exports the eventlog to a TFTP server with an IP address of 192.168.77.14 with a filename of eventlog-exported:

```
console> eventlog export 192.168.77.14 eventlog-exported
```

The following is sample output from the **eventlog show** command:

```
console> eventlog show
Aug 18 14:10:01 ce-1 pang[26997]: /mnt/vol15       mounted_cmdb      active    ce-1
352        24         327        92.90
Aug 18 14:10:01 ce-1 pang[26997]: /mnt/vol25       mounted           active    ce-1
352        26         325        92.42
Aug 18 14:10:01 ce-1 pang[26997]: /mnt/vol27       mounted           active    ce-1
352        18         334        0.00
Aug 18 14:10:01 ce-1 pang[26997]: /mnt/vol29       mounted           active    ce-1
352        15         336        95.61
Aug 18 14:10:01 ce-1 pang[26997]: /mnt/vol33       mounted           active    ce-1
259        11         247        95.59
Aug 18 14:14:01 ce-1 pang[26997]: Interface eth5 is down
Aug 18 14:14:01 ce-1 pang[26997]: Interface eth4 is down
Aug 18 14:14:01 ce-1 pang[26997]: Interface eth7 is down
Aug 18 14:14:01 ce-1 pang[26997]: Interface eth6 is down
```

```
Aug 18 14:14:01 ce-1 pang[26997]: all bridges (2) are down , will go to disable mode
Aug 18 14:14:01 ce-1 pang[26997]: Operational state has been set to disabled
Aug 18 14:14:01 ce-1 pang[26997]: detected  major:    operational mode has been changed
Aug 18 14:14:23 mg-1 snmpd[17343]: cluster has been degraded
Aug 18 14:14:30 ce-3 pang[7806]: volume             state      availability owner
total      free       used       usage
Aug 18 14:14:30 ce-3 pang[7806]: /mnt/vol8          mounted     active    ce-3  352
18         333        94.73
Aug 18 14:14:30 ce-3 pang[7806]: /mnt/vol10         mounted     active    ce-3  352
18         334        94.87
Aug 18 14:14:30 ce-3 pang[7806]: /mnt/vol12         mounted     active    ce-3  352
17         334        94.94
Aug 18 14:14:30 ce-3 pang[7806]: /mnt/vol14         mounted     active    ce-3  352
22         330        93.67
<output omitted>
```

Note In a Cisco VDS TC installation that uses the Cisco Blade Servers, you may see “cluster has been enabled” followed by “cluster has been degraded” SNMP messages in the eventlog. When the Cisco VDS TC system does not receive traffic on the cache engine interfaces, it believes there may be a problem with the interfaces. In an attempt to “fix” this perceived problem, the system disables and enables the application, causing the “cluster has been enabled” and the “cluster has been degraded” messages to appear in the logs.

The following is sample output from the **eventlog tail** command:

```
console> eventlog tail
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol11        mounted          active   ce-1
352        17         334        94.96
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol15        mounted_cmdb      active  ce-1
352        24         327        92.92
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol25        mounted           active  ce-1
352        26         325        92.47
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol27        mounted           active  ce-1
352        17         334        94.92
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol29        mounted           active  ce-1
352        15         337        95.66
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol33        mounted           active  ce-1
259        11         247        95.63
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol35        mounted           active  ce-1
259        10         248        95.77
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol37        mounted           active  ce-1
259        10         248        95.78
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol40        mounted           active  ce-1
259        11         247        0.00
Aug 18 14:19:17 ce-2 pang[7769]: Current leader is me (#5053416419#ce-2)! Num members = 2
```

exit

To exit any mode or close an active CLI session use the **exit** command. In Enable mode, this command returns the user to Regular mode. In Regular mode, this command terminates the session and the user is logged out of the CLI session.

exit

Syntax Description

This command has no arguments or keywords.

Command Modes Regular mode, Enable mode, and Configuration mode

Examples The following example shows how to exit the current session:

```
console> exit
```

help

To display the CLI commands that are available in the current mode, with a short description of each command, use the **help** command. For example, if you enter the **help** command in Regular mode, then the command displays only the commands that are available in Regular mode with a short description of each command.

help

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode, Enable mode, and Configuration mode

Examples The following example displays the list of CLI commands that are available in Regular mode:

```
console> ?
arp                 Show arp table
current_cli_users   Show currently logged in cli users
direction           Calculate seen subnets on interface
dmesg               Display dmesg
dstat               Display IO statistics
enable              Enter privileged mode
eventlog            Event log commands
exit                Exit current mode
help                Commands description
ifconfig            Display interface(s)
iostat              Display IO statistics
jumbo               Send jumbo echo messages
ping                Send echo messages
show                Show run-time information
traceroute          Print the route packets take to network host
```

ifconfig

To display the details of the interfaces, use the **ifconfig** command.

ifconfig

Syntax Description This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **ifconfig** command:

```
console> ifconfig
bond0     Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:10953821962 errors:0 dropped:0 overruns:2 frame:0
          TX packets:3299545150 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:4849898198811 (4625223.3 Mb)  TX bytes:630496892356 (601288.6 Mb)

bond0.50  Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          inet addr:10.11.12.1  Bcast:10.11.12.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:7255222368 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2934064031 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2714904229216 (2589134.4 Mb)  TX bytes:262983669899 (250800.7 Mb)

bond0.60  Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          inet addr:10.11.18.1  Bcast:10.11.18.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:108073349 errors:0 dropped:0 overruns:0 frame:0
          TX packets:89034578 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:62844043700 (59932.7 Mb)  TX bytes:5951898176 (5676.1 Mb)

bond0.60: Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          inet addr:192.168.128.100  Bcast:192.168.128.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1

bond0:1   Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          inet addr:10.56.194.29  Bcast:10.56.195.255  Mask:255.255.254.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1

eth0      Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:7508473145 errors:0 dropped:0 overruns:2 frame:0
          TX packets:3299264321 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2947208349095 (2810676.9 Mb)  TX bytes:630467083736 (601260.2 Mb)

eth1      Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:3445348817 errors:0 dropped:0 overruns:0 frame:0
          TX packets:280829 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1902689849716 (1814546.4 Mb)  TX bytes:29808620 (28.4 Mb)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:2309190022 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2309190022 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:868439099052 (828208.0 Mb)  TX bytes:868439099052 (828208.0 Mb)
```

iostat

To report CPU statistics and input/output statistics for devices and partitions, use the **iostat** command.

iostat [**-t** *interval* [**-k** *count*]]

Syntax Description

-t *interval*	The amount of time, in seconds between each report. The default is 5 seconds.
-k *count*	Used in conjunction with the interval parameter. If the *count* parameter is specified, the count determines the number of reports generated at the specified interval. If the interval parameter is specified without the *count* parameter, the **iostat** command generates reports continuously until you press **Ctrl-C**.

Command Modes

Regular mode and Enable mode

Examples

The following example generates two I/O statistics reports two seconds apart:

```
console> iostat -t 2 -k 4
Linux 2.6.21-affined-8-default (mg-1)   04/22/09

Time: 13:18:42
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          11.29    0.03    6.02    1.38    0.00   81.28

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.50 120.18  0.62  9.44   98.11 1050.22    49.06   525.11   114.11     0.68
67.44   5.78   5.82

Time: 13:18:46
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           9.48    0.00    6.11    1.68    0.00   82.73

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.00  41.79  0.00 10.95    0.00  429.85     0.00   214.93    39.27     0.13
11.82   6.36   6.97

Time: 13:18:50
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          10.38    0.00    6.25    1.25    0.00   82.12

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.00  27.50  0.00  8.75    0.00  298.00     0.00   149.00    34.06     0.10
11.66   6.51   5.70

Time: 13:18:54
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           9.05    0.00    6.05    1.00    0.00   83.90
```

```
Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.00  30.67  0.00  7.98    0.00  313.22     0.00   156.61    39.25     0.10
12.00   5.25   4.19
```

jumbo

To send jumbo echo messages, use the **jumbo** command.

jumbo [**-c** *counter*] [**-I** {*IP_address* | *interface*}] *destination*

Syntax Description

-c *counter*	The number of times the request is generated.
-I {*IP_address* | *interface*}	The interface IP address or interface name from which the echo requests are sent.
destination	The destination to which the jumbo echo message will be sent.

Command Modes

Regular mode and Enable mode

Examples

The following example sends jumbo echo messages to 192.168.5.117 sourced from the eth0 interface:

```
cconsole> jumbo -I eth0 192.168.5.117
PING 192.168.5.117 (192.168.5.117) from 192.168.5.117 eth0: 8972(9000) bytes of data.
8980 bytes from 192.168.5.117: icmp_seq=1 ttl=64 time=0.043 ms
8980 bytes from 192.168.5.117: icmp_seq=2 ttl=64 time=0.024 ms
8980 bytes from 192.168.5.117: icmp_seq=3 ttl=64 time=0.028 ms
8980 bytes from 192.168.5.117: icmp_seq=4 ttl=64 time=0.033 ms
8980 bytes from 192.168.5.117: icmp_seq=5 ttl=64 time=0.039 ms

--- 192.168.5.117 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3998ms
rtt min/avg/max/mdev = 0.024/0.033/0.043/0.008 ms

--- 192.168.0.202 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3997ms
rtt min/avg/max/mdev = 0.030/0.037/0.049/0.007 ms
```

The following example sends two jumbo echo messages to 192.168.3.170 sourced from the eth0 interface:

```
console> jumbo -c 2 -I eth0 192.168.3.170
PING 192.168.3.170 (192.168.3.170) from 192.168.5.117 eth0: 8972(9000) bytes of data.
From 192.168.5.117 icmp_seq=1 Frag needed and DF set (mtu = 1500)
From 192.168.5.117 icmp_seq=1 Frag needed and DF set (mtu = 1500)

--- 192.168.3.170 ping statistics ---
0 packets transmitted, 0 received, +2 errors
```

ping

To diagnose basic network connectivity, use the **ping** command. The **ping** command uses the ICMP protocol mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway, and displays the round-trip time for the echo response to arrive back to the server on which the command was issued.

ping [**-c** *counter*] [**-I** {*IP_address* | *interface*}] *destination*

Syntax Description

-c *counter*	The number of ICMP echo requests to be sent to the destination address. If you do not specify the number of echo requests to send, the ping will continue until you press **Ctrl-C**.
-I {*IP_address* | *interface*}	The interface IP address or interface name from which the pings are sent.
destination	The destination to which the ping messages will be sent. This value can be an IP address or a hostname.

Command Modes

Regular mode and Enable mode

Usage Guidelines

To abort the ping command, press **Ctrl-C**.

Examples

The following example pings the destination at 192.168.5.117 sourced from the eth0 interface until Ctrl-C is pressed:

```
console> ping -I eth0 192.168.5.117
PING 192.168.5.117 (192.168.5.117) from 192.168.5.117 eth0: 56(84) bytes of data.
64 bytes from 192.168.5.117: icmp_seq=1 ttl=64 time=0.019 ms
64 bytes from 192.168.5.117: icmp_seq=2 ttl=64 time=0.012 ms
64 bytes from 192.168.5.117: icmp_seq=3 ttl=64 time=0.015 ms
64 bytes from 192.168.5.117: icmp_seq=4 ttl=64 time=0.012 ms
64 bytes from 192.168.5.117: icmp_seq=5 ttl=64 time=0.015 ms
64 bytes from 192.168.5.117: icmp_seq=6 ttl=64 time=0.014 ms
64 bytes from 192.168.5.117: icmp_seq=7 ttl=64 time=0.021 ms
64 bytes from 192.168.5.117: icmp_seq=8 ttl=64 time=0.011 ms
64 bytes from 192.168.5.117: icmp_seq=9 ttl=64 time=0.012 ms
64 bytes from 192.168.5.117: icmp_seq=10 ttl=64 time=0.012 ms
64 bytes from 192.168.5.117: icmp_seq=11 ttl=64 time=0.014 ms
64 bytes from 192.168.5.117: icmp_seq=12 ttl=64 time=0.012 ms

--- 192.168.5.117 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11013ms
rtt min/avg/max/mdev = 0.011/0.014/0.021/0.003 ms
```

show

To display run-time information related to the operational environment of VDS TC, use the **show** command.

show {connectivity | detection_rules | dstat | eth_status | eventlog | leader | process | status | systemid | time | uptime | version | volumes}

Syntax Description

cluster-bus-ip	Display cluster communication bus status
connectivity	Display the iSCSI connectivity information.
detection_rules	Displays detection rules.
dstat	Displays I/O statistics.
eth_status	Displays the eth status.
eventlog	Displays the platform event log. This command displays the same results as the **eventlog show** command.
leader	Displays the hostname of the current cluster leader. The cluster leader manages resources used by VDS TC.
process	Displays the status of VDS TC components (caching application, spread, and apache) as they run on the platform. The output of this command is relevant for maintenance engineers.
status	Displays the cluster administrative and application status.
systemid	Displays the system serial number.
time	Displays the system date and time.
uptime	Displays the uptime for the appliance.
version	Displays the software version.
volumes	Displays the mounted volumes.

Note Additional parameters for this command are available in Enable mode. Refer to the **show** command in the Enable Mode Commands section for a description of these parameters.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **show eth_status** command:

```
console> show eth_status
Blade     eth0    eth1    eth2    eth3    eth4    eth5    eth6    eth7    eth8    eth9
eth10   eth11   eth12   eth13
ce-1       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-2       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-3       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-4       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-5       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-6       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
```

```
ce-7       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-8       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-9       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-10      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-11      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-12      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-13      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-14      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-15      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
ce-16      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP      UP
```

The following is sample output from the **show connectivity** command:

```
console# show connectivity
Blade   Active iSCSI sessions
ce-1    9
ce-2    9
ce-3    9
ce-4    9
ce-5    9
ce-6    9
ce-7    9
ce-8    9
ce-9    9
ce-10   9
ce-11   9
ce-12   9
ce-13   9
ce-14   9
ce-15   9
ce-16   9
Do you want to see more detailed information ? [y|n]  :y

Please enter blade number [1-16] :3
10.11.14.101
10.11.14.104
10.11.14.105
10.11.14.102
10.11.14.103
10.11.14.106
10.11.14.107
10.11.14.109
10.11.14.108
```

The following is sample output from the **show eventlog** command:

```
console> show eventlog
Aug 18 14:16:16 ce-2 pang[7769]: /mnt/vol23         mounted     active    ce-2  352
33         318        90.44
Aug 18 14:17:01 ce-1 pang[26997]: volume             state      availability owner
total      free       used       usage
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol7         mounted           active  ce-1
352        20         331        94.04
```

```
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol9         mounted           active  ce-1
352        18         333        94.72
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol11        mounted           active  ce-1
352        17         334        94.96
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol15        mounted_cmdb      active ce-1
352        24         327        92.92
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol25        mounted           active ce-1
352        26         325        92.47
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol27        mounted           active ce-1
352        17         334        94.92
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol29        mounted           active ce-1
352        15         337        95.66
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol33        mounted           active ce-1
259        11         247        95.63
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol35        mounted           active ce-1
259        10         248        95.77
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol37        mounted           active ce-1
259        10         248        95.78
Aug 18 14:17:01 ce-1 pang[26997]: /mnt/vol40        mounted           active ce-1
259        11         247        0.00
Aug 18 14:19:17 ce-2 pang[7769]: Current leader is me (#5053416419#ce-2)! Num members = 2
<output omitted>
```

The following is sample output from the **show leader** command:

```
console> show leader
ce-2
```

The following is sample output from the **show process** command:

```
console> show process
wwwrun    1954     1  0 Feb27 ?        01:00:30 python ubview.py --logfile
/opt/pang/mgmt/django/run/ubview.log --pidfile /opt/pang/mgmt/django/run/ubview.pid
--reactor epoll
root      2709     1  0 Feb19 ?        00:00:43 /opt/pang/mgmt//avalon/sbin/snmptrapd -f
-Osq -Ls u -c /opt/pang/mgmt//avalon/sbin/snmptrapd.conf 10.11.12.1
wwwrun    3309  6877  0 22:03 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun    3310  6877  0 22:03 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
ntp       6669     1  0 Feb25 ?        00:10:29 /usr/sbin/ntpd -p /var/run/ntp/ntpd.pid -n
-g -u ntp:ntp -U 0 -i /var/lib/ntp -c /etc/ntp.conf
admin     6690  6689  0 Feb19 pts/1    00:00:00 pang_cli
root      6877     1  0 Feb25 ?        00:01:32 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun    8225  6877  0 22:05 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
admin    11721 11718  0 21:55 pts/9    00:00:00 -pang_cli
wwwrun   12317  6877  0 21:55 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun   12490  6877  0 22:06 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
root     14479     1  2 Feb19 ?        19:55:00 /opt/pang/mgmt/bin/monitor
wwwrun   14861  6877  0 22:07 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun   14862  6877  0 22:07 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
spread   16390     1  7 Feb24 ?        2-14:24:48 /usr/bin/spread -n mg-1 -c
/etc/spread.conf
wwwrun   18636  6877  0 21:57 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
root     21280     1  0 Feb27 ?        03:26:14 /opt/pang/mgmt/avalon/sbin/snmpd -f -A -LF
e /opt/pang/mgmt/avalon/var/log/snmpd.log -LS c u 0.0.0.0
```

```
wwwrun   24200  6877  0 21:49 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun   24321  6877  0 21:49 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
```

The following is sample output from the **show status** command:

```
console> show status
Cluster state: degraded
Blade Slot  Status     Operational state   Device state   Administrative state
ce-1       powered on      enabled              starting             unlocked
ce-2       powered on      enabled              started              unlocked
ce-3       powered on      enabled              started              unlocked
ce-4       powered on        N/A                stopped              unlocked
```

The following is sample output from the **show systemid** command:

```
console> show systemid
*H6L1K3J
7GB9C4J
3GB9C4J
9GB9C4J
4GB9C4J
```

The following is sample output from the **show time** command:

```
console> show time
Mon Mar 31 2014 23:37:32 GMT+0200
```

The following is sample output from the **show uptime** command:

```
console> show uptime
cluster              4 days, 6h:34m:38s
ce-1                 4 days, 6h:35m:08s
ce-2                 4 days, 6h:35m:07s
ce-3                 4 days, 6h:35m:05s
ce-4                 4 days, 6h:35m:04s
ce-5                 4 days, 6h:35m:03s
ce-6                 4 days, 6h:35m:02s
ce-7                 4 days, 6h:35m:01s
ce-8                 4 days, 6h:35m:00s
ce-9                 4 days, 6h:34m:58s
ce-10                4 days, 6h:34m:57s
ce-11                4 days, 6h:34m:56s
ce-12                4 days, 6h:34m:55s
ce-13                4 days, 6h:34m:54s
ce-14                4 days, 6h:34m:52s
ce-15                4 days, 6h:34m:51s
ce-16                4 days, 6h:34m:50s
```

The following is sample output from the **show version** command:

```
console> show version
show version
VDS-TC Transparent Caching           cli version 5.7.3b54
management           VDS-TC Transparent Caching mgmt software version 5.7.3b54
ce-1                 5.7.3b54                LLPF Version LLPF_05.7.3b53-54
ce-2                 5.7.3b54                LLPF Version LLPF_05.7.3b53-54
ce-3                 5.7.3b54                LLPF Version LLPF_05.7.3b53-54
ce-4                 5.7.3b54                LLPF Version LLPF_05.7.3b53-54
ce-5                 5.7.3b54                LLPF Version LLPF_05.7.3b53-54
ce-6                 5.7.3b54                LLPF Version LLPF_05.7.3b53-54
ce-7                 5.7.3b54                LLPF Version LLPF_05.7.3b53-54
ce-8                 5.7.3b54                LLPF Version LLPF_05.7.3b53-54
ce-9                 5.7.3b54                LLPF Version LLPF_05.7.3b53-54
```

```
ce-10                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-11                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-12                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-13                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-14                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-15                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-16                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
```

The following is sample output from the **show volumes** command:

```
console> show volumes
Licensed volumes : 120
Volume name          State                Owner
/mnt/vol1            mounted              ce-1
/mnt/vol2            mounted              ce-5
/mnt/vol3            mounted              ce-12
/mnt/vol4            mounted              ce-10
/mnt/vol5            mounted              ce-2
/mnt/vol6            mounted              ce-10
/mnt/vol7            mounted              ce-1
/mnt/vol8            mounted              ce-8
/mnt/vol9            mounted              ce-1
/mnt/vol10           mounted              ce-13
/mnt/vol11           mounted              ce-6
/mnt/vol12           mounted              ce-11
/mnt/vol13           mounted              ce-2
/mnt/vol14           mounted              ce-9
/mnt/vol15           mounted              ce-2
/mnt/vol16           mounted              ce-14
/mnt/vol17           mounted              ce-3
/mnt/vol18           mounted              ce-12
/mnt/vol19           mounted              ce-4
/mnt/vol20           mounted              ce-9
/mnt/vol21           mounted              ce-2
/mnt/vol22           mounted              ce-14
/mnt/vol23           mounted              ce-7
/mnt/vol24           mounted              ce-4
/mnt/vol25           mounted              ce-3
/mnt/vol26           mounted              ce-6
/mnt/vol27           mounted              ce-3
/mnt/vol28           mounted              ce-14
/mnt/vol29           mounted              ce-5
/mnt/vol30           mounted              ce-13
/mnt/vol31           mounted              ce-4
/mnt/vol32           mounted              ce-10
/mnt/vol33           mounted              ce-3
/mnt/vol34           mounted              ce-15
/mnt/vol35           mounted              ce-8
/mnt/vol36           mounted              ce-1
/mnt/vol37           mounted              ce-5
/mnt/vol38           mounted              ce-7
/mnt/vol39           mounted              ce-5
/mnt/vol40           mounted              ce-12
/mnt/vol41           mounted              ce-6
/mnt/vol42           mounted              ce-14
/mnt/vol43           mounted              ce-4
<output omitted>
```

traceroute

To discover the IP routes that packets will actually take when traveling to their destination, use the **traceroute** command. The **traceroute** command tracks the route of a packet across a TCP/IP network on its way to a given host. It utilizes the IP protocol time to live (TTL) field and attempts to elicit an ICMP TIME_EXCEEDED response from each gateway along the path to the destination.

traceroute [**-n**] *destination*

Syntax Description

-n	Forces the traceroute command to avoid mapping IP addresses to host names when displaying the output.
destination	The name or IP address of the destination.

Command Modes

Regular mode and Enable mode

Examples

The following example shows a traceroute to the destination 213.25.17.65:

```
console> traceroute 213.25.17.65
traceroute to 213.25.17.65 (213.25.17.65), 30 hops max, 40 byte packets
 1  192.168.5.1 (192.168.5.1)  1.301 ms   1.279 ms   1.292 ms
 2  10.1.2.253 (10.1.2.253)  0.201 ms   0.237 ms   0.177 ms
 3  212.150.66.65 (212.150.66.65)  0.990 ms   1.045 ms   0.821 ms
 4  212.150.6.137 (212.150.6.137)  4.394 ms   4.411 ms   4.267 ms
 5  gig0-1-gw1.thc.nv.net.il (212.143.200.33)  4.321 ms   4.281 ms   4.927 ms
 6  coresw1-hfa-gw1.thc.nv.net.il (212.143.200.1)  5.146 ms   5.578 ms   5.657 ms
 7  gi0-1-0.gw1.hfa.nv.net.il (212.143.8.193)  5.027 ms   6.523 ms   6.389 ms
 8  pos1-3-0.gw1.lnd.nv.net.il (212.143.12.81)  98.327 ms   102.262 ms   101.749 ms
 9  ten3-1.brdr1.lnd.nv.net.il (212.143.14.137)  86.039 ms   89.152 ms   89.150 ms
10  10.50.1.1 (10.50.1.1)  94.458 ms   90.483 ms   90.919 ms
11  ldn-b2-link.telia.net (213.248.72.125)  88.340 ms   88.606 ms   99.483 ms
12  ldn-tch-i1-link.telia.net (80.91.250.217)  95.796 ms   91.841 ms   92.334 ms
13  ldn-b5-link.telia.net (80.91.250.209)  95.562 ms   92.757 ms   93.582 ms
14  * * *
15  xe-1-0-1-0.ffttr2.Frankfurt.opentransit.net (193.251.129.45)  190.518 ms   184.738 ms
180.753 ms
16  * * *
17  do.wro-ar3.z.wro-r1.tpnet.pl (213.25.5.154)  138.287 ms   134.848 ms
do.wro-ar3.z.wro-r2.tpnet.pl (213.25.12.154)  132.107 ms
18  z-easy-com.wro-ar1.tpnet.pl (80.50.233.62)  138.366 ms   134.411 ms   135.860 ms
19  z-easy-com.wro-ar1.tpnet.pl (80.50.233.62)  137.239 ms   133.277 ms   134.191 ms
```

The following example shows a traceroute to the destination 213.25.17.65 without resolving hostnames:

```
console> traceroute -n 213.25.17.65
traceroute to 213.25.17.65 (213.25.17.65), 30 hops max, 40 byte packets
 1  192.168.0.2 (192.168.0.2)  1.307 ms   2.133 ms   2.832 ms
 2  10.1.2.253 (10.1.2.253)  0.472 ms   0.477 ms   0.438 ms
 3  212.150.66.65 (212.150.66.65)  1.150 ms   1.391 ms   2.616 ms
 4  212.150.6.137 (212.150.6.137)  6.146 ms *   4.927 ms
 5  212.143.200.33 (212.143.200.33)  5.112 ms   5.374 ms   5.457 ms
 6  212.143.200.1 (212.143.200.1)  6.348 ms   7.459 ms   6.200 ms
 7  212.143.8.196 (212.143.8.196)  6.269 ms   6.208 ms   6.420 ms
 8  212.143.12.57 (212.143.12.57)  75.146 ms   74.670 ms   74.082 ms
```

```
 9  146.188.55.61 (146.188.55.61)  68.022 ms   68.586 ms   69.261 ms
10  158.43.150.145 (158.43.150.145)  76.673 ms *   76.009 ms
11  146.188.15.245 (146.188.15.245)  68.753 ms *   67.996 ms
12  146.188.4.42 (146.188.4.42)  70.045 ms *   71.441 ms
13  * * 146.188.112.82 (146.188.112.82)  69.605 ms
14  * 193.251.129.81 (193.251.129.81)  75.391 ms   74.884 ms
15  193.251.129.41 (193.251.129.41)  93.854 ms   169.094 ms   165.100 ms
16  * * *
17  213.25.5.154 (213.25.5.154)  118.610 ms 213.25.12.154 (213.25.12.154)  122.907 ms
122.927 ms
18  80.50.233.62 (80.50.233.62)  218.868 ms   218.119 ms *
19  80.50.233.62 (80.50.233.62)  123.890 ms   125.167 ms   124.627 ms
```

Enable Mode Commands

This section describes the commands that are available in Enable mode. The Regular mode commands are also available in Enable mode. The Enable Mode Commands table lists the commands that are described in this section.

You must have the Enable password to access Enable mode. *See* Switching from Regular Mode to Enable Mode, page 7-4.

Note All of the CLI commands work with both IPv4 and IPv6.

Note All commands and their parameters are case sensitive.

Table 9-2 Enable Mode Commands

Command	Description
access	Manages system access parameters
apache_restart	Restarts the apache server
arp	Displays the ARP table
cache	Manages additional cache operations
config	Enters Configuration mode
current_cli_users	Displays the list of VDS TC admin users who are currently logged into the VDS TC CLI
detection_rules	Manages detection rules configuration
direction	Calculates the visible subnets on the interface
dmesg	Displays the message buffer
eventlog	Provides access to event log operations
exit	Exits the current mode
help	Displays the list of available commands for the mode you are in when you execute the command
ifconfig	Displays the interface(s)

Table 9-2 Enable Mode Commands (continued)

Command	Description
iostat	Displays extended I/O statistics
jumbo	Sends jumbo echo messages
license	Manages the system license
oper	Provides access to system management operations
ping	Sends echo messages
reset	Resets management services
show	Displays run-time information
system	System information
traceroute	Displays the route used by the packet to reach its destination
upgrade	Downloads and installs a software image file
vlan	Adds and removes VLANs from the interface

access

To manage system access parameters, use the **access** command.

access {**enable-password** | **idle-session-timeout** *value* | **user-password**}

Syntax Description

enable-password	Sets the Enable mode password. After you enter the command, the system will prompt you to enter a new password and re-enter the password to confirm it. The new password should be at least four characters long.
idle-session-timeout *value*	Sets the timeout, in seconds, after which the Telnet or SSH session is terminated, both for the Enable mode and the Regular mode commands. The default value is 0 seconds, which disables the timeout feature.
user-password	Sets the Regular mode command user password. After you enter this command, the system will prompt you to first verify the existing password. You will then be prompted to enter a new password and re-enter the password to confirm it. The new password should be at least four characters long The system will verify the password against a set of rules to ensure that the password is complex enough.

Note The default Enable mode password is set during system installation and defaults to the system-id. You can view the system-id using the **show systemid** command in Regular mode. It is strongly recommended that you change the default Enable mode password immediately after the initial installation.

Command Modes

Enable mode

Examples

The following is an example of the dialog that occurs when using the **access enable-password** command:

```
console# access enable-password
New password:
Re-enter new password:
```

The following example sets the idle session timeout value to 32768 seconds:

```
console# access idle-session-timeout 32768
```

The following is an example of the dialog that occurs when using the **access user-password** command:

```
console# access user-password
Changing password for admin.
Old Password:
New Password:
Bad password: too simple
New Password:
Reenter New Password:
Password changed.
```

apache_restart

To restart the apache server, use the **apache_restart** command.

apache_restart

Syntax Description

This command has no arguments or keywords.

Command Modes

Enable mode

Examples

The following is sample output from the **ifconfig** command:

```
console# apache_restart
Restarting httpd2 (SIGHUP)               done
```

arp

To display the ARP table use the **arp** command.

arp

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **arp** command:

```
console> arp
ce-16                   ether   d4:8c:b5:4d:a9:32   C                     bond0.50
10.56.194.1             ether   00:00:0c:07:ac:c8   C                     bond0
10.11.18.204            ether   00:80:e5:36:0e:60   C                     bond0.60
10.11.18.206            ether   00:80:e5:2f:ed:4e   C                     bond0.60
ce-15                   ether   a4:4c:11:29:b8:9c   C                     bond0.50
10.11.18.208            ether   00:80:e5:36:40:54   C                     bond0.60
10.56.194.99            ether   3c:08:f6:60:71:92   C                     bond0
ce-1                    ether   a4:93:4c:aa:46:e8   C                     bond0.50
ce-8                    ether   d4:8c:b5:4d:a7:30   C                     bond0.50
ce-7                    ether   a4:93:4c:aa:8e:f6   C                     bond0.50
10.11.18.205            ether   00:80:e5:36:10:a6   C                     bond0.60
10.11.18.203            ether   00:80:e5:2f:f8:ae   C                     bond0.60
10.11.18.207            ether   00:80:e5:36:10:c0   C                     bond0.60
10.11.18.200            ether   00:80:e5:2f:d8:3c   C                     bond0.60
ce-4                    ether   d4:8c:b5:4d:a7:4c   C                     bond0.50
ce-2                    ether   d4:8c:b5:4d:de:04   C                     bond0.50
10.11.18.209            ether   00:80:e5:36:47:42   C                     bond0.60
ce-3                    ether   c4:64:13:39:a8:ec   C                     bond0.50
10.56.194.117           ether   3c:08:f6:60:8b:b6   C                     bond0
ce-9                    ether   d4:8c:b5:4d:98:80   C                     bond0.50
ce-11                   ether   a4:93:4c:aa:6b:8a   C                     bond0.50
ce-14                   ether   fc:99:47:49:c4:ee   C                     bond0.50
10.11.18.201            ether   00:80:e5:36:02:b2   C                     bond0.60
ce-5                    ether   d4:8c:b5:4d:a5:1e   C                     bond0.50
ce-12                   ether   d4:8c:b5:4d:cf:4e   C                     bond0.50
ce-13                   ether   50:57:a8:e1:a1:96   C                     bond0.50
10.11.18.202            ether   00:80:e5:2f:f5:5c   C                     bond0.60
ce-10                   ether   a4:4c:11:2a:0c:c0   C                     bond0.50
ce-6                    ether   d4:8c:b5:4d:b7:d8   C                     bond0.50
```

cache

To manage the cache parameters, use the **cache** command.

cache {**active_sessions** *IP_address* | **black_list** {**add** *hash_id* | **dump** | **export** *tftp_server filename* | **remove** *hash_id*} | **hash** *hash_id* | **list** {**display** | **export** *tftp_server filename* | **short**} | **remove hash** *hash_id* | **summary** | **sync** | **volume** {**activate** | **deactivate** | **insert** | **remove** *volume_id* | **remove_content**}}

Syntax Description

active_sessions *IP_address*	Displays the active sessions for the IP address referenced with the *IP_address* parameter.
black_list {**add** *hash_id* | **dump** | **export** *tftp_server filename* | **remove** *hash_id*}	• **black_list add** *hash_id:* Adds a file to the black list that has a hash ID that matches the *hash_id* parameter. • **black_list dump**: Displays (dumps) the entire black list. • **black_list export** *tftp_server filename*: Exports the black list to a TFTP server, where *tftp_server* is the hostname or IP address of the TFTP server and *filename* is the name of the to export to. **Note:** The file to which the content is exported must already exist, and must have write access to all. • **remove** *hash_id*: Removes a file from the black list that has a hash ID that matches the *hash_id* parameter.

hash *hash_id*	Dumps the files metadata that has a hash id that matches the *hash_id* parameter.
list {**display** | **export** *tftp_server filename* | **short**}	• **list display**: Displays a full list of cache content. • **list export** *tftp_server filename*: Exports the cache content to a TFTP server, where *tftp_server* is the hostname or IP address of the TFTP server and *filename* is the name of the to export to. • **list short**: Displays the top 1000 Least Recently Used (LRU) cached hash IDs. This process can take about 2 minutes. To interrupt the process, press **Ctrl-C**.
remove hash *hash_id*	Removes a file from cache whose hash ID value matches the *hash_id* parameter. The *hash_id* parameter should match a hash ID that exists in the system cache. For a list of hash IDs stored in the system, use the cache list command.
summary	Displays the CMDB statistics summary.
sync	Verifies and synchronizes the cache metadata. The platform is fully accessible during this process. Note that synching the cache can take a few hours.
volume {**activate** *volume_id* | **deactivate** *volume_id* | **insert** | **remove** *volume_id* | **remove_content** *volume_id*}	The **cache volume** command manipulates the cache file system volumes. The **cache volume** commands are mainly used for maintenance purposes, usually for hard drive maintenance. You can view the volumes that can be used for these commands using the **show volumes** command. • **volume activate**: Activates a cache volume. When you enter this command, you will be prompted to enter which volume number you want to activate. • **volume deactivate**: Deactivates a cache volume. When you enter this command, you will be prompted to enter which volume number you want to deactivate. • **volume insert**: Searches for new disks to add to the system. • **volume remove** *volume_id*: Removes all hash IDs associated with the volume specified with the *volume_id* parameter from within the configuration management database (CMDB), so that the system will no longer cache these hash IDs. This command removes ALL information cached on this volume from the CMDB. This is a non-reversible process. • **volume remove_content**: Removes content from a volume. When you enter this command, you will be prompted to enter the volume number from which you want to remove content.

Command Modes

Enable mode

Examples

The following are examples of managing the black list:

- Adds a file to the black list, based on the hash ID:

```
console# cache black_list add AE7E21FB0CA2DD7464A562E74064248E9B790057
The specified hash was inserted in a black list
```

```
console#
console# cache black_list add 6827AC55B43B1B0BAB58FC9F9E7D6B05EF71FDCD
The specified hash was inserted in a black list
```

- Displays the contents of the black list:

```
console# cache black_list dump
HASH                                     PROTOCOL          SIZE            AGE
6827AC55B43B1B0BAB58FC9F9E7D6B05EF71FDCD P2P_SIGNATURE_NA    0               0
AE7E21FB0CA2DD7464A562E74064248E9B790057 P2P_SIGNATURE_NA    0               0
```

- Exports the black list to a TFTP server with in IP address of 192.168.14.26 and a filename of black-list:

```
console# cache black_list export 192.168.14.26 black-list
```

- Removes a file from the black list based on the hash ID:

```
console# cache black_list remove 6827AC55B43B1B0BAB58FC9F9E7D6B05EF71FDCD
The specified hash was deleted from a black list
```

The following example displays the metadata for a file based on its hash ID:

```
console# cache hash BCBBAF652BFEAAE3E11C3F279608A1FB7A337DCD
This operation might take some time.(^C to interrupt)
.
.
.
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
HTTP 126B32237A1EDF3508B35B49642945F2E97FB8E2 ST:WATCHED HITS:6 IPS:1 VL:14 SZ:6299870
MP:0 FF:yes FS:1252582601 LS:125267625
1 CO:6 MB FF:100.00 PFF:0.00 checksum 91B6735FB6557E3C959B278C8C0AC2323DD84CC8
Hits: 6
First seen: Thu Sep 10 11:36:41 2009
Last access: Fri Sep 11 13:37:31 2009
Volume id:  14
Cached File size:  6299870
Max piece:  0
Full File:  yes
Full File Size:  6299870
BT_last_start: 0
BT_last_end:   6299870
max known bt piece size 0
cache out in MB 6
File fill factor 100.00
total full pieces 0 (0.00)
HTTP checksum is 91B6735FB6557E3C959B278C8C0AC2323DD84CC8

piece bit mask 0x2
RANGES -------------- RANGES
```

The following are examples of working with the cache content options:

- Displays the cache content:

```
console# cache list display
This operation might take some time as the whole storage-index is scanned.(^C to
interrupt)
HASH                                     PROTOCOL   SIZE            AGE
D35D8E3D8EEE3BF82D7B8F783FD4D2380A216C67 HTTP       6276762         42
208FD9F7E7C42473291C503931EF22269657285A HTTP       6286497         30
5A88AC47F6B636E5FF6C462D9DE15691A53B26C3 HTTP       6297586         29
26D26351170461CDD3B5680ED0417254C4FA43C1 HTTP       6291394         26
```

```
86BA4AC4E0E818C18BD2B9040D7A5E3F19E70776 HTTP       6276664         43
9ADB98FC073B05866EFFBC1159522E9E42261669 HTTP       6295830         43
.
```

- Exports the full list of hash IDs to a TFTP server with an IP address of 192.168.5.117:

```
console# cache list export 192.168.5.117 cache-list
```

- Displays the top 1000 least recently used hash IDs. This operation can take about 2 minutes to complete. Press **CTRL-C** to interrupt the process.

```
console# cache list short
This operation might take some time as the whole storage-index is scanned.(^C to
interrupt)
HASH                                     PROTOCOL   SIZE            AGE
D35D8E3D8EEE3BF82D7B8F783FD4D2380A216C67 HTTP       6276762         42
208FD9F7E7C42473291C503931EF22269657285A HTTP       6286497         30
5A88AC47F6B636E5FF6C462D9DE15691A53B26C3 HTTP       6297586         29
26D26351170461CDD3B5680ED0417254C4FA43C1 HTTP       6291394         26
86BA4AC4E0E818C18BD2B9040D7A5E3F19E70776 HTTP       6276664         43
9ADB98FC073B05866EFFBC1159522E9E42261669 HTTP       6295830         43
94611C230B77D73205BC3090FA2A6104BAEE7990 HTTP       6297222         31
<output omitted>
```

The following example removes a hash ID that is stored in the cache:

```
console# cache remove hash F753B1C31107981BC86D87CF5F7B9EEFD5F5A28B
The specified hash will be deleted in a few minutes
```

The following example starts a cache verification and synchronization process:

```
console# cache sync
Proceeding cache to metadata sync. Some data in the cache might be lost.
Are you sure (y/n)? y
Starting cache synchronization...
```

The following examples show working with volumes:

- Activates a specific volume:

```
console# cache volume activate
Licensed volumes: 120
Please enter volume number <1-120> 4
```

- Deactivates a specific volume:

```
console# cache volume deactivate
Licensed volumes: 120
Please enter volume number <1-120> 4
```

- Removes a specific volume from the CMDB:

```
console# cache volume remove 4
Are you sure? This will remove all hashes from volume 4.
[yes|no] no

Removing volume 4 has been cancelled
```

config

From Configuration mode you can make platform configuration changes. To enter Configuration mode, use the **config** command.

config

Note For a list of commands available in Configuration mode, *see* Configuration Mode Commands.

Syntax Description This command has no arguments or keywords.

Command Modes Enable mode

Usage Guidelines Changes that you make in Configuration mode are stored but are only implemented when you enter the **apply** command. Use the **exit** command to exit Configuration mode and return to Enable mode.

Examples The following example enters Configuration mode:

```
console# config
configuration#
```

current_cli_users

To display the admin users who are currently logged on, use the **current_cli_users** command.

current_cli_users

Note You must be logged on using an admin username and password to view the output from this command. This command only displays admin users that are logged directly into the VDS TC CLI. It does not display users that have used the sudo command from the Linux CLI to access the VDS TC CLI.

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode and Enable mode

Examples The following example displays a list of users currently logged on to the system:

```
console# current_cli_users
admin    pts/1        Apr  4 21:32 (10.21.150.101)
```

detection_rules

To view detection rules deployment dates and versions and to export the detection rules configurations, use the **detection_rules** command.

detection_rules {**export_groups** *tftp_server filename* | **export_signatures** *tftp_server filename*| **show**}

Syntax Description

export_groups *tftp_server filename*	Exports the group configuration to a TFTP server, where the TFTP server IP address or hostname and filename are specified with the *tftp_server* and *filename* parameters.
export_signatures *tftp_server filename*	Exports the signature configuration to a TFTP server, where the TFTP server IP address or hostname and filename are specified with the *tftp_server* and *filename* parameters.
show	Displays the detection rules versions and deployment dates.

Command Modes

Enable mode

Examples

The following example exports the groups and the signature files to the TFTP server on the VDS TC system:

```
console# detection_rules export_groups 127.0.0.1 _groups
console# detection_rules export_signatures 127.0.0.1 _signatures
```

direction

To calculate the visible subnets on the specified interface, use the **direction** command.

direction *interface_name*

Syntax Description

interface_name	The interface for which you want to display the subnets, for example eth0.

Command Modes

Regular mode and Enable mode

Examples

The following sample displays the visible subnets on the interface eth0 using the **direction** command:

```
console> direction eth0
tcpdump: WARNING: eth0: no IPv4 address assigned
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
1000 packets captured
```

```
1001 packets received by filter
0 packets dropped by kernel
    889 10.11.12.0/24
     64 10.11.18.0/24
      6 10.11.80.0/24
     40 10.56.10.0/24
      1 64.101.47.0/24
```

dmesg

To displays the message buffer of the kernel, use the **dmesg** command.

dmesg

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **dmesg** command:

```
console> dmesg
usb 3-1.1: Manufacturer: ATEN International Co. Ltd
usb 3-1.1: configuration #1 chosen from 1 choice
input: ATEN International Co. Ltd CS1716A V1.0.098 as /class/input/input9
input: USB HID v1.00 Keyboard [ATEN International Co. Ltd CS1716A V1.0.098] on
usb-0000:00:1d.1-1.1
input: ATEN International Co. Ltd CS1716A V1.0.098 as /class/input/input10
input: USB HID v1.00 Device [ATEN International Co. Ltd CS1716A V1.0.098] on
usb-0000:00:1d.1-1.1
input: ATEN International Co. Ltd CS1716A V1.0.098 as /class/input/input11
input: USB HID v1.10 Mouse [ATEN International Co. Ltd CS1716A V1.0.098] on
usb-0000:00:1d.1-1.1
usb 1-5.2: USB disconnect, address 6
usb 1-5.1: new high speed USB device using ehci_hcd and address 8
usb 1-5.1: new device found, idVendor=0781, idProduct=5406
usb 1-5.1: new device strings: Mfr=1, Product=2, SerialNumber=3
usb 1-5.1: Product: U3 Cruzer Micro
usb 1-5.1: Manufacturer: SanDisk
usb 1-5.1: SerialNumber: 40549102FB103472
usb 1-5.1: configuration #1 chosen from 1 choice
scsi3 : SCSI emulation for USB Mass Storage devices
usb-storage: device found at 8
usb-storage: waiting for device to settle before scanning
scsi 3:0:0:0: Direct-Access     SanDisk  U3 Cruzer Micro  8.02 PQ: 0 ANSI: 0 CCS
sd 3:0:0:0: Attached scsi removable disk sdc
sd 3:0:0:0: Attached scsi generic sg3 type 0
usb-storage: device scan complete
usb 1-5.1: USB disconnect, address 8
usb 1-4: new high speed USB device using ehci_hcd and address 9
usb 1-4: new device found, idVendor=0781, idProduct=5406
usb 1-4: new device strings: Mfr=1, Product=2, SerialNumber=3
usb 1-4: Product: U3 Cruzer Micro
usb 1-4: Manufacturer: SanDisk
usb 1-4: SerialNumber: 40549102FB103472
<output omitted>
```

eventlog

The event log lists all of the log messages sent to the system log by all of the VDS TC service components (applications, CLI and SNMP). To view or export the content of the event log, use the **eventlog** command. The eventlog command that is available in Enable mode includes additional parameters that are not available in Regular mode. The options are used to duplicate the event log information to an external syslog server.

eventlog {**date** *date*| **export** *tftp_server filename* | **forward** | **show** | **stop** | **tail**}

Syntax Description

date *date*	Displays the eventlog for the date specified with the *date* parameter. The *date* parameter is in the format of DD-MM-YYYY. Press **q** to exit the eventlog and return to the console# prompt.
export *tftp_server filename*	Exports the content of the event log to the TFTP server with the IP address or hostname specified with the *tftp_server* parameter.
forward	Starts eventlog forwarding to a previously configured syslog server.
show	Displays the event log entries. Press **q** to exit the eventlog and return to the console# prompt.
stop	Stops eventlog forwarding to a previously configured syslog server.
tail	Displays the online event log entries.

Command Modes

Regular mode and Enable mode. The **start** and **stop** options are available only in Enable mode.

Usage Guidelines

To add an external syslog server for VDS TC to use, add the following statements to the system configuration:

```
<mgmt-config>
    <external_syslog_ip>ip_address</external_syslog_ip>
</mgmt-config>
```

The *ip_address* parameter specifies the IP address of the external syslog server to use.

Examples

The following example starts forwarding event log messages to the syslog server that is configured in the cluster_conf.xml file:

```
console# eventlog forward
```

The following example displays the eventlog for March 31, 2014:

```
console# eventlog date 31-03-2014
Mar 31 23:58:57 ce-14 pang[3648]: /mnt/vol16       mounted_cmdb      active    ce-14
259        123        135        52.27      0
Mar 31 23:58:57 ce-14 pang[3648]: /mnt/vol27       mounted           active    ce-14
259        123        135        52.24      0
Mar 31 23:58:57 ce-14 pang[3648]: /mnt/vol30       mounted           active    ce-14
259        123        135        52.38      0
Mar 31 23:59:04 ce-3 pang[4065]: Volume /dev/sdac3 (id 4) was turned ON
```

```
Mar 31 23:59:08 ce-10 pang[19976]: volume             state      availability owner
total      free       used       usage      total_writes
Mar 31 23:59:08 ce-10 pang[19976]: /mnt/vol4        mounted         active   ce-10
259        134        124        48.09      0
Mar 31 23:59:08 ce-10 pang[19976]: /mnt/vol5        mounted         active   ce-10
259        116        142        55.03      0
Mar 31 23:59:08 ce-10 pang[19976]: /mnt/vol20       mounted         active   ce-10
259        119        139        53.96      0
Mar 31 23:59:08 ce-10 pang[19976]: /mnt/vol28       mounted_cmdb     active   ce-10
259        117        141        54.63      0
Mar 31 23:59:08 ce-10 pang[19976]: /mnt/vol38       mounted         active   ce-10
259        117        141        54.54      0
Mar 31 23:59:08 ce-10 pang[19976]: /mnt/vol65       mounted         active   ce-10
259        121        137        53.09      0
Mar 31 23:59:08 ce-10 pang[19976]: /mnt/vol80       mounted         active   ce-10
259        114        144        55.74      0
Mar 31 23:59:36 ce-4 pang[25424]: volume             state      availability owner
total      free       used       usage      total_writes
Mar 31 23:59:36 ce-4 pang[25424]: /mnt/vol13       mounted_cmdb      active   ce-4
259        132        127        49.00      0
Mar 31 23:59:36 ce-4 pang[25424]: /mnt/vol21       mounted          active   ce-4
259        133        125        48.32      0
```

The following example stops forwarding event log messages to the configured syslog server:

```
console# eventlog stop
```

exit

To exit any mode or close an active CLI session use the **exit** command. In Enable mode, this command returns the user to Regular mode. In Regular mode, this command terminates the session and the user is logged out of the CLI session.

exit

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode, Enable mode, and Configuration mode

Examples

The following example shows how to exit the current session:

```
console> exit
```

help

To display the CLI commands that are available in the current mode, with a short description of each command, use the **help** command. For example, if you enter the **help** command in Enable mode, then the command displays only the commands that are available in Enable mode with a short description of each command.

help

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode, Enable mode, and Configuration mode

Examples The following example displays the list of CLI commands that are available in Enable mode:

```
console# ?
access               Manage system access parameters
apache_restart       Restart apache
arp                  Show arp table
cache                Cache operations
config               Enter configuration mode
current_cli_users  Show currently logged in cli users
detection_rules      Manage detection rules configuration
direction            Calculate seen subnets on interface (IPv4 traffic only)
dmesg                Display dmesg
eventlog             Event log commands
exit                 Exit current mode
help                 Commands description
ifconfig             Display interface(s)
iostat               Display IO statistics
jumbo                Send jumbo echo messages
license              Manage system license
oper                 System management operations
ping                 Send echo messages
reset                Reset management service
show                 Show run-time information
system               System information
traceroute           Print the route packets take to network host
upgrade              Download and install software image file
vlan                 VLAN operations
```

ifconfig

To display the details of the interfaces, use the **ifconfig** command.

ifconfig

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode and Enable mode

Examples The following is sample output from the **ifconfig** command:

```
console# ifconfig
bond0     Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:10962344879 errors:0 dropped:0 overruns:2 frame:0
          TX packets:3302539313 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:4853456069280 (4628616.3 Mb)  TX bytes:630780014950 (601558.6 Mb)
```

```
bond0.50  Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          inet addr:10.11.12.1  Bcast:10.11.12.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:7260803088 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2936818470 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2716849041988 (2590989.1 Mb)  TX bytes:263226693190 (251032.5 Mb)

bond0.60  Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          inet addr:10.11.18.1  Bcast:10.11.18.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:108311195 errors:0 dropped:0 overruns:0 frame:0
          TX packets:89236667 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:62980460642 (60062.8 Mb)  TX bytes:5966422490 (5690.0 Mb)

bond0.60: Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          inet addr:192.168.128.100  Bcast:192.168.128.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1

bond0:1   Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          inet addr:10.56.194.29  Bcast:10.56.195.255  Mask:255.255.254.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1

eth0      Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:7514337189 errors:0 dropped:0 overruns:2 frame:0
          TX packets:3302258483 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2949375048070 (2812743.2 Mb)  TX bytes:630750206235 (601530.2 Mb)

eth1      Link encap:Ethernet  HWaddr D4:8C:B5:4D:B8:E2
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:3448007688 errors:0 dropped:0 overruns:0 frame:0
          TX packets:280829 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1904081020284 (1815873.1 Mb)  TX bytes:29808620 (28.4 Mb)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:2311026801 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2311026801 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:869080640236 (828819.8 Mb)  TX bytes:869080640236 (828819.8 Mb)
```

iostat

To report CPU statistics and input/output statistics for devices and partitions, use the **iostat** command.

iostat [**-t** *interval* [**-k** *count*]]

Syntax Description

-t *interval*	The amount of time, in seconds between each report. The default is 5 seconds.
-k *count*	Used in conjunction with the interval parameter. If the *count* parameter is specified, the count determines the number of reports generated at the specified interval. If the interval parameter is specified without the *count* parameter, the **iostat** command generates reports continuously until you press **Ctrl-C**.

Command Modes

Regular mode and Enable mode

Examples

The following example generates two I/O statistics reports four seconds apart:

```
console# iostat -t 2 -k 4
Linux 2.6.21-affined-8-default (mg-1)   04/22/09

Time: 13:18:42
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          11.29    0.03    6.02    1.38    0.00   81.28

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.50 120.18  0.62  9.44   98.11 1050.22    49.06   525.11   114.11     0.68
67.44   5.78   5.82

Time: 13:18:46
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           9.48    0.00    6.11    1.68    0.00   82.73

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.00  41.79  0.00 10.95    0.00  429.85     0.00   214.93    39.27     0.13
11.82   6.36   6.97

Time: 13:18:50
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          10.38    0.00    6.25    1.25    0.00   82.12

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.00  27.50  0.00  8.75    0.00  298.00     0.00   149.00    34.06     0.10
11.66   6.51   5.70

Time: 13:18:54
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           9.05    0.00    6.05    1.00    0.00   83.90

Device:    rrqm/s wrqm/s   r/s   w/s  rsec/s  wsec/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda          0.00  30.67  0.00  7.98    0.00  313.22     0.00   156.61    39.25     0.10
12.00   5.25   4.19
```

jumbo

To send jumbo echo messages, use the **jumbo** command.

jumbo [**-c** *counter*] [**-I** {*IP_address* | *interface*}] *destination*

Syntax Description

-c *counter*	The number of times the request is generated.
-I {*IP_address* | *interface*}	The interface IP address or interface name from which the echo requests are sent.
destination	The destination to which the jumbo echo message will be sent.

Command Modes

Regular mode and Enable mode

Examples

The following example sends jumbo echo messages to 192.168.5.117 sourced from the eth0 interface:

```
console# jumbo -I eth0 192.168.5.117
PING 192.168.5.117 (192.168.5.117) from 192.168.5.117 eth0: 8972(9000) bytes of data.
8980 bytes from 192.168.5.117: icmp_seq=1 ttl=64 time=0.043 ms
8980 bytes from 192.168.5.117: icmp_seq=2 ttl=64 time=0.024 ms
8980 bytes from 192.168.5.117: icmp_seq=3 ttl=64 time=0.028 ms
8980 bytes from 192.168.5.117: icmp_seq=4 ttl=64 time=0.033 ms
8980 bytes from 192.168.5.117: icmp_seq=5 ttl=64 time=0.039 ms

--- 192.168.5.117 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3998ms
rtt min/avg/max/mdev = 0.024/0.033/0.043/0.008 ms
```

The following example sends two jumbo echo messages to 192.168.3.170 sourced from the eth0 interface:

```
console> jumbo -c 2 -I eth0 192.168.3.170
PING 192.168.3.170 (192.168.3.170) from 192.168.5.117 eth0: 8972(9000) bytes of data.
From 192.168.5.117 icmp_seq=1 Frag needed and DF set (mtu = 1500)
From 192.168.5.117 icmp_seq=1 Frag needed and DF set (mtu = 1500)

--- 192.168.3.170 ping statistics ---
0 packets transmitted, 0 received, +2 errors
```

license

To manage VDS-TC system license, use the **license** command. This license controls operational parameters, such as the supported protocols and features, and the maximum cache bandwidth.

license {**activate** | **import** *tftp_server filename* | **show**}

Syntax Description

activate	Activates the system license.
import *tftp_server filename*	Imports a license from the TFTP server where the server name or IP address and file location are specified with the *tftp_server* and *filename* parameters.
show	Displays the currently licensed operational parameters.

Command Modes

Enable mode

Examples

The following example activates the installed license:

```
console# license activate
Licensed chassis serial number: H6L1K3J
Licensed chassis serial number: 7GB9C4J
Licensed chassis serial number: 3GB9C4J
Licensed chassis serial number: 9GB9C4J
Licensed chassis serial number: 4GB9C4J
ENumber of blades:               17
EDK enabled:                            1
Bittorent enabled:                      1
Kazaa enabled:                          1
Gnutella enabled:                       1
Ares enabled:                           1
Http enabled:                           1
Pando enabled:                          1
Thunder enabled:                        0
Smartfilter enabled:                    0
Netflix enabled:                        1
Silverlight enabled:                    1
Storage volumes:                        120
Controllers:                            1
CDR logs:                               1
Service Detection:                      1
web Cache enabled:                      0
N_PLUS_K enabled:                       0
Evaluation ends on:                     14-5-2014
Max bandwidth:                          unlimited
Max forwarding:                         unlimited
Are you sure that you want to activate this license ? (y/n)? y
Activating license...
```

The following example imports the license from the TFTP server with an IP address of 10.1.1.65:

```
console# license import 10.1.1.65 License.xml
Licensed chassis serial number: H6L1K3J
Licensed chassis serial number: 7GB9C4J
Licensed chassis serial number: 3GB9C4J
Licensed chassis serial number: 9GB9C4J
Licensed chassis serial number: 4GB9C4J
Number of blades:               17
EDK enabled:                            1
Bittorent enabled:                      1
Kazaa enabled:                          1
Gnutella enabled:                       1
Ares enabled:                           1
Http enabled:                           1
Pando enabled:                          1
Thunder enabled:                        0
```

```
Smartfilter enabled:                    0
Netflix enabled:                        1
Silverlight enabled:                    1
Storage volumes:                        120
Controllers:                            1
CDR logs:                               1
Service Detection:                      1
web Cache enabled:                      0
N_PLUS_K enabled:                       0
Evaluation ends on:                     14-5-2014
Max bandwidth:                          unlimited
Max forwarding:                         unlimited
```

The following example displays the installed license operational parameters:

```
console# license show
Licensed chassis serial number: H6L1K3J
Licensed chassis serial number: 7GB9C4J
Licensed chassis serial number: 3GB9C4J
Licensed chassis serial number: 9GB9C4J
Licensed chassis serial number: 4GB9C4J
Number of blades:               17
EDK enabled:                            1
Bittorent enabled:                      1
Kazaa enabled:                          1
Gnutella enabled:                       1
Ares enabled:                           1
Http enabled:                           1
Pando enabled:                          1
Thunder enabled:                        0
Smartfilter enabled:                    0
Netflix enabled:                        1
Silverlight enabled:                    1
Storage volumes:                        120
Controllers:                            1
CDR logs:                               1
Service Detection:                      1
web Cache enabled:                      0
N_PLUS_K enabled:                       0
Evaluation ends on:                     14-5-2014
Max bandwidth:                          unlimited
Max forwarding:                         unlimited
```

oper server *server_number*

From Server mode you can start, stop, or restart an individual server, and set the log level of the server. In Server mode the CLI server prompt appears: oper server <server number> # (for example: oper server 1#). To enter Server mode, use the **oper server** command.

oper server *server_number*

Syntax Description

server_number	The number of the server you want to manage.

Note For a list of commands available in Server mode, *see* Server Mode Commands, page 9-54.

Command Modes Enable mode

oper service

To control the running state of the platform, including starting and stopping the platform software and all its services, use the **oper service** command.

oper service {**start** | **stop**}

Syntax Description

start	Starts the VDS TC software and services.
stop	Stops the VDS TC software and services

Command Modes Enable mode

Examples The following example stops the VDS TC software and its services:

```
console# oper service stop
Are you sure (y/n)? y
Stopping service
console# exit
console> show status
Cluster state: disabled

Blade Slot   Status   Operational state   Device state   Administrative state
ce-1         powered on       enabled       started              unlocked
ce-2         powered on       N/A           N/A                  unlocked
ce-3         powered on       enabled       started              unlocked
ce-4         powered on       enabled       started              unlocked
ce-5         powered on       enabled       started              unlocked
ce-6         powered on       enabled       started              unlocked
ce-7         powered on       enabled       started              unlocked
ce-8         powered on       enabled       started              unlocked
ce-9         powered on       enabled       started              unlocked
ce-10        powered on       enabled       started              unlocked
ce-11        powered on       enabled       started              unlocked
ce-12        powered on       enabled       started              unlocked
```

The following example stars the VDS TC software and its services:

```
console# oper service start
Starting service
console# show status
Operational state    Device state         Administrative state
 enabled              started              unlocked
```

ping

To diagnose basic network connectivity, use the **ping** command. The **ping** command uses the ICMP protocol mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway, and displays the round-trip time for the echo response to arrive back to the server on which the command was issued.

Note To abort the ping command, press **Ctrl-C**.

ping [**-c** *counter*] [**-I** {*IP_address* | *interface*}] *destination*

Syntax Description

-c *counter*	The number of icmp echo requests to be sent to the destination address. If you do not specify the number of echo requests to send, the ping will continue until you press **Ctrl-C**.
-I {*IP_address* | *interface*}	The interface IP address or interface name from which the pings are sent.
destination	The destination to which the ping messages will be sent. This value can be an IP address or a hostname.

Command Modes

Regular mode and Enable mode

Usage Guidelines

To abort the ping command, press **Ctrl-C**.

Examples

The following example pings the destination at 192.168.0.202 sourced from the eth0 interface until Ctrl-C is pressed:

```
console> ping -I eth0 192.168.0.202
PING 192.168.0.202 (192.168.0.202) from 192.168.0.202 eth0: 56(84) bytes of data.
64 bytes from 192.168.0.202: icmp_seq=1 ttl=64 time=0.023 ms
64 bytes from 192.168.0.202: icmp_seq=2 ttl=64 time=0.035 ms
64 bytes from 192.168.0.202: icmp_seq=3 ttl=64 time=0.028 ms
64 bytes from 192.168.0.202: icmp_seq=4 ttl=64 time=0.020 ms
64 bytes from 192.168.0.202: icmp_seq=5 ttl=64 time=0.029 ms
64 bytes from 192.168.0.202: icmp_seq=6 ttl=64 time=0.030 ms
64 bytes from 192.168.0.202: icmp_seq=7 ttl=64 time=0.032 ms
64 bytes from 192.168.0.202: icmp_seq=8 ttl=64 time=0.029 ms
64 bytes from 192.168.0.202: icmp_seq=9 ttl=64 time=0.030 ms
64 bytes from 192.168.0.202: icmp_seq=10 ttl=64 time=0.030 ms
--- 192.168.0.202 ping statistics ---
10 packets transmitted, 0 received, 0% packet loss, time 19023ms
rtt min/avg/max/mdev = 0.019/0.027/0.037/0.006 ms
```

reset

To reset the management services, use the **reset** command.

Note Resetting the management services will disconnect *your* current administration session, and you will have to login again.

reset

Syntax Description

This command has no arguments or keywords.

Command Modes

Enable mode

Examples

The following example resets the management service and all of its services:

```
console# reset
Are you sure (y/n)? y
.
.Connection terminated
```

show

To display run-time information related to the operational environment of VDS TC, use the **show** command. The Enable mode show command includes all of the parameters available in Regular mode (*see* show, page 9-12) and the following additional parameters:

show {**config** | **license**}

Syntax Description

config	Displays the running configuration.
license	Displays the system license information.

Note Additional parameters for this command are available in Regular mode. Refer to the **show** command in the Regular Mode Commands section for a description of these parameters.

Command Modes

Enable mode

Examples

The following is sample output from the **show config** command:

```
console# show config
<cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='./cluster_conf.xsd'>
  <mgmt-config>
  <ipaddr>10.56.194.29</ipaddr><!-- Change it according to the customer network
-->
```

```
  <netmask>255.255.254.0</netmask><!-- Change it according to the customer network
-->
    <ip6addr>2001:470:470:3::85/56</ip6addr><!-- Optional - Change it according to the
customer network                          -->
  <default-gw>10.56.194.1</default-gw><!-- Change it according to the customer network
-->
    <default6-gw>2001:470:470:3::84</default6-gw><!-- Optional - Change it according to
the customer network                          -->
    <nameserver>8.8.8.8</nameserver><!-- Recommended atleast one DNS server, Change it
according to the customer network -->
    <nameserver>0.0.0.0</nameserver><!-- Optional - Change it according to the customer
network                          -->
    <alert-email>tech-support@domain.com</alert-email><!-- Optional - Change it according
to the customer network                          -->
    <site_name>Cisco VDS-TC-16S+1</site_name><!-- Influence on the title of the VDS-TC
Manager window and in the system mail      -->
  </mgmt-config>
  <web-config>
    <controller>
      <ip>10.11.18.200</ip>
    </controller>
    <controller>
      <ip>10.11.18.201</ip>
    </controller>
    <controller>
      <ip>10.11.18.202</ip>
    </controller>
    <controller>
      <ip>10.11.18.203</ip>
    </controller>
    <controller>
      <ip>10.11.18.204</ip>
    </controller>
    <controller>
      <ip>10.11.18.205</ip>
    </controller>
    <controller>
      <ip>10.11.18.206</ip>
    </controller>
    <controller>
      <ip>10.11.18.207</ip>
    </controller>
    <controller>
      <ip>10.11.18.208</ip>
    </controller>
    <controller>
      <ip>10.11.18.209</ip>
    </controller>
  </web-config>
  <common>
<ntp>
      <server-ip>0.pool.ntp.org</server-ip>
      <timezone>GMT+2</timezone>
    </ntp>
    <snmp>
      <trap-ip>10.11.12.1</trap-ip>
      <snmp-read-community>gdcbhv</snmp-read-community>
      <snmp-write-community>nkppui</snmp-write-community>
      <snmp-trap-community>ffff</snmp-trap-community>
    </snmp>
    <service>
      <protocols>
        <enable-bittorent>1</enable-bittorent>
        <enable-edk>1</enable-edk>
```

```
        <enable-http>1</enable-http>
        <enable-ares>1</enable-ares>
        <enable_cache_out_port>1</enable_cache_out_port>
      </protocols>
      <net>
        <fwd-mode>BOUNCING</fwd-mode>
        <bounce id='0'></bounce>
   <subnet_range_per_link name='a'><!-- Change it according to the customer network
-->
          <cidr_range>10.0.0.0/8</cidr_range>
          <cidr_range>2000::/3</cidr_range>
        </subnet_range_per_link>
        <management_switch_ip1>10.11.12.201</management_switch_ip1>
        <management_switch_ip2>10.11.12.202</management_switch_ip2>
      </net>
      <policy></policy>
      <cache-memory-DB></cache-memory-DB>
      <memory></memory>
    </service>
  </common>
.....
```

(output omitted)

The following is example output from the **show license** command:

```
console# show license
Licensed chassis serial number: FCH1611V0G8
Licensed chassis serial number: FCH161976UV
Licensed chassis serial number: FCH161578HM
Licensed chassis serial number: FCH16207KN9
Licensed chassis serial number: FCH161971S7
Licensed chassis serial number: FCH162073XL
Licensed chassis serial number: FCH162079J7
Licensed chassis serial number: FCH162079EZ
Licensed chassis serial number: FCH161979LM
Licensed chassis serial number: FCH16327X7R
Licensed chassis serial number: FCH161971MP
Licensed chassis serial number: FCH163770MR
Licensed chassis serial number: FCH16337CXS
Licensed chassis serial number: FCH163170AE
Licensed chassis serial number: FCH163673F8
Licensed chassis serial number: FCH16367DSD
Licensed chassis serial number: FCH163170HE
Number of blades:               16
EDK enabled:                            1
Bittorent enabled:                      1
Kazaa enabled:                          1
Gnutella enabled:                       1
Ares enabled:                           1
Http enabled:                           1
Pando enabled:                          1
Thunder enabled:                        0
Smartfilter enabled:                    0
Netflix enabled:                        1
Silverlight enabled:                    1
Storage volumes:                        120
Controllers:                            1
CDR logs:                               1
Service Detection:                      1
web Cache enabled:                      0
N_PLUS_K enabled:                       0
Evaluation ends on:                     14-5-2014
Max bandwidth:                          unlimited
Max forwarding:                         48000 Mbps
```

system

To export system log or system statistics information, use the **system** command. This command contains the following parameters:

system {**logs export** | **statistics export**}

Syntax Description

logs export	Exports the system logs to a TFTP server
statistics export	Exports the system statistics to a TFTP server

Command Modes

Enable mode

traceroute

To discover the IP routes that packets will actually take when traveling to their destination, use the **traceroute** command. The **traceroute** command tracks the route of a packet across a TCP/IP network on its way to a given host. It utilizes the IP protocol time to live (TTL) field and attempts to elicit an ICMP TIME_EXCEEDED response from each gateway along the path to the destination.

traceroute [**-n**] *destination*

Syntax Description

-n	Forces the traceroute command to avoid mapping IP addresses to host names when displaying the output.
destination	The name or IP address of the destination.

Command Modes

Regular mode and Enable mode

Examples

The following example shows a traceroute to the destination 213.25.17.65:

```
console> traceroute 213.25.17.65
traceroute to 213.25.17.65 (213.25.17.65), 30 hops max, 40 byte packets
 1  192.168.0.2 (192.168.0.2)  1.316 ms   1.485 ms   1.292 ms
 2  10.1.2.253 (10.1.2.253)  0.656 ms   0.372 ms   0.747 ms
 3  212.150.66.65 (212.150.66.65)  1.481 ms   1.210 ms   1.044 ms
 4  212.150.6.137 (212.150.6.137)  5.517 ms   6.398 ms   6.574 ms
 5  gig0-1-gw1.thc.nv.net (212.143.200.33)  10.862 ms   10.842 ms   10.668 ms
 6  * coresw1-hfa-gw1.thc.nv.net (212.143.200.1)  6.721 ms   6.895 ms
 7  teng2-1-0.gw2.hfa.nv.net (212.143.8.196)  5.959 ms   9.117 ms   8.460 ms
 8  pos0-1-1.brdr2.lnd.nv.net (212.143.12.57)  77.423 ms *   75.177 ms
 9  GigabitEthernet3-1.GW1.LND9.ALTER.NET (146.188.55.61)  73.425 ms   69.143 ms 70.365 ms
10  * so-3-0-0.CR2.LND9.ALTER.NET (158.43.150.145)  157.824 ms   153.850 ms
11  so-0-3-0.XT2.LND2.ALTER.NET (146.188.15.245)  69.099 ms   69.834 ms *
12  GigabitEthernet1-0-0.BR1.LND18.ALTER.NET (146.188.4.42)  73.002 ms   72.350 ms
78.342 ms
```

```
13  GE.LND.opentransit.net (146.188.112.82)  70.164 ms   69.356 ms   69.688 ms
14  tengige0-8-0-0.lontr1.London.opentransit.net (193.251.129.81)  80.167 ms   80.134 ms
79.107 ms
15  xe-0-0-1-0.ffttr2.Frankfurt.opentransit.net (193.251.129.41)  92.546 ms   95.034 ms
94.282 ms
16  * * *
17  do.wro-ar3.z.wro-r1.tpnet.pl (213.25.5.154)  120.296 ms do.wro-ar3.z.wro-r2.tpnet.pl
(213.25.12.154)  134.187 ms   130.254 ms
18  * z-easy-com.wro-ar1.tpnet.pl (80.50.233.62)  236.345 ms   235.563 ms
19  z-easy-com.wro-ar1.tpnet.pl (80.50.233.62)  124.651 ms   125.124 ms   126.872 ms
```

upgrade

To upgrade the software version of VDS TC, use the **upgrade** command. You can upgrade all servers, just the management server, or a specific server.

upgrade {all *tftp_server file* | management *tftp_server file* | server *server_no tftp_server file}*

Syntax Description

all *tftp_server file*	Downloads and installs software image file for all servers. • *tftp_server*: This parameter is the hostname or IP address of the TFTP server. This server must be accessible from the VDS TC platform on which you are running the **upgrade** command. • *file*: This parameter is the name of the file containing the software version package received from Cisco. **Note:** If the TFTP server is running on one of the Cisco servers, the upgrade command attempts to retrieve the file from the /tftpboot folder.
management *tftp_server file*	Downloads and installs software image file for the management server. • *tftp_server*:This parameter is the hostname or IP address of the TFTP server. This server must be accessible from the VDS TC platform on which you are running the **upgrade** command. • *file*: This parameter is the name of the file containing the software version package received from Cisco. **Note:** If the TFTP server is running on one of the Cisco servers, the upgrade command attempts to retrieve the file from the /tftpboot folder.
server *server_no* **t***ftp_server file*	Downloads and installs software image file for the management server. • *server_no*: The specific VDS TC server number on which you want to run the **upgrade** command • *tftp_server*:This parameter is the hostname or IP address of the TFTP server. This server must be accessible from the VDS TC platform on which you are running the **upgrade** command. • *file*: This parameter is the name of the file containing the software version package received from Cisco. **Note:** If the TFTP server is running on one of the Cisco servers, the upgrade command attempts to retrieve the file from the /tftpboot folder.

Note If you do not have root access, the file can also be uploaded to the /tftpboot folder from the Config Files Management option of VDS TC Manager. For more information, *see* "Upload Files" in the *Cisco Videoscape Distribution Suite Transparent Caching Manager User Guide*. After uploading the file, you can upgrade using the **upgrade** command at the VDS TC CLI.

Command Modes

Enable mode

Note For complete details on upgrading your VDS TC software, see the *Cisco Videoscape Distribution Suite Transparent Caching Application Upgrade Guide* at http://www.cisco.com/c/dam/en/us/td/docs/video/videoscape/distribution_suite/vds/v5_7_3/VDS-TC_5.7.3_app_upgrade_guide.pdf.

vlan

To add or remove a VLAN to or from an interface, use the **vlan** command

vlan {**add** *interface_name vlan_id ip* [*mask*] | **remove** *interface_name vlan_id*}

Syntax Description

add *interface_name vlan_id ip* [*mask*]	Adds the new vlan specified with the *vlan_id* parameter to the interface indicated with the *interface_name* parameter. You must specify an IP address for the interface.
remove *interface_name vlan_id*	Removes the vlan specified with the *vlan_id* parameter from the interface indicated with the *interface_name* parameter.

Command Modes

Enable mode

Examples

The following example adds vlan 10 to eth0 with an IP address 10.11.12.15 and subnet mask of 255.255.255.0:

```
console# vlan add eth0 10 10.11.12.15 255.255.255.0
Set name-type for VLAN subsystem. Should be visible in /proc/net/vlan/config
Added VLAN with VID == 10 to IF -:eth0:-
interface eth0.10 is up
```

The following example removes vlan 10 from eth0:

```
console# vlan remove eth0 10
Removed VLAN -:eth0.10:-
eth0.10 removed
```

Configuration Mode Commands

This section describes the commands that are available in Configuration mode. The Configuration Mode Commands table lists the commands that are described in this section.

Table 9-3 Configuration Mode Commands

Command	Description
apply	Applies the configuration changes requested until now.
cluster-bus-ip	Configure cluster bus type
diff	Shows the pending configuration changes.
discard	Discards the pending changes.
display	Displays the current configuration.
exit	Exits Configuration mode.
export	Exports the cluster configuration to the TFTP server.
help	Displays the command syntax for each configuration command.
import	Imports the cluster configuration from the TFTP server.
network	Configures the management network interface.
restore	Restores the last good configuration.
time	Sets the system date and time.

apply

To apply the configuration changes requested until now, use the **apply** command. This command immediately applies the configuration changes to the live platform.

apply

Syntax Description

This command has no arguments or keywords.

Command Modes

Configuration mode

Usage Guidelines

If there are no configuration changes to apply, this command will return the message “Configurations are identical.”.

Examples

The following is sample output from the **apply** command:

```
configuration# apply
```

```
applying configuration...
Configuration applied
```

cluster-bus-ip

The cluster-bus-ip command is used to configure whether spread communication among cluster members is set to broadcast or multicast. For a VDS TC 5.1.1 installation, this should always be set to broadcast.

Note This configuration should only be changed by Cisco engineers. Changing this parameter can cause serious stability issues.

diff

To display the proposed configuration changes, use the **diff** command.

diff

Syntax Description

This command has no arguments or keywords.

Command Modes

Configuration mode

Usage Guidelines

When you enter the **diff** command, the new configuration parameters are indicated by a plus (**+**) sign as the first character on the line, while the current configuration parameters are indicated by a minus (**-**) sign as the first character on the line. If there are no proposed configuration changes to display, the command returns the message “Configurations are identical”.

Examples

The following is sample output from the **diff** command:

```
configuration# diff
  <?xml version="1.0" encoding="UTF-8" standalone="no" ?>

  <cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='cluster_conf.xsd'>
    <mgmt-config>
      <ipaddr>192.168.0.202</ipaddr>
      <netmask>255.255.255.0</netmask>
      <default-gw>192.168.0.2</default-gw>
      <nameserver>10.1.1.235</nameserver>
      <alert-email>support@cisco.com</alert-email>
      <site_name>192.168.0.202</site_name>
      <external_syslog_ip>127.0.0.1</external_syslog_ip>
    </mgmt-config>
    <web-config></web-config>
         <snmp>
        <trap-ip>aa.bb.cc.dd</trap-ip>
        <snmp-read-community>gdcbhv</snmp-read-community>
        <snmp-write-community>nkppui</snmp-write-community>
```

```
      <snmp-trap-community>nkppui</snmp-trap-community>
    </snmp>
    <service>
      <protocols>
        <enable-bittorent>1</enable-bittorent>
        <enable-edk>1</enable-edk>
        <enable-http>1</enable-http>
        <enable-ares>1</enable-ares>
                </protocols>
      <net>
        <fwd-mode>PROMISC</fwd-mode>
        <bridge id='0'>
          <interface-world>iff2</interface-world>
          <interface-isp>iff1</interface-isp>
        </bridge>
      </net>
    </service>
  </common>
  <blades>1</blades>
  <blade id='1'>
    <cache-engine>
      <network></network>
           </cache-engine>
  </blade>
</cluster>
```

discard

To discard any configuration change requested since entering Configuration mode, use the **discard** command.

discard

Syntax Description

This command has no arguments or keywords.

Command Modes

Configuration mode

Examples

The following is sample output from the **discard** command:

```
configuration# discard
```

display

To display the current configuration, use the **display** command.

display

Syntax Description

This command has no arguments or keywords.

Command Modes

Configuration mode

Examples

The following is sample output from the **display** command:

```
configuration# display
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>

<cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='cluster_conf.xsd'>
    <mgmt-config>
        <ipaddr>10.56.194.29</ipaddr>
        <netmask>255.255.254.0</netmask>
        <default-gw>10.56.194.1</default-gw>
        <nameserver>8.8.8.8</nameserver>
        <site_name>CISCO UCS Grid 2.101_test</site_name>
        <external_syslog_ip>10.1.1.85</external_syslog_ip>
       <enable-equipment-traps>1</enable-equipment-traps>

        <access_lists>
            <black_access_list>
                <access_entry>10.1.1.65</access_entry>
            </black_access_list>
        </access_lists>

    </mgmt-config>
    <web-config>
        <controller>
            <ip>10.11.18.200</ip>
        </controller>
        <controller>
            <ip>10.11.18.201</ip>
        </controller>
        <controller>
            <ip>10.11.18.202</ip>
        </controller>
        <controller>
            <ip>10.11.18.203</ip>
        </controller>
        <controller>
            <ip>10.11.18.204</ip>
        </controller>
        <controller>
            <ip>10.11.18.205</ip>
        </controller>
        <controller>
            <ip>10.11.18.206</ip>
        </controller>
        <controller>
            <ip>10.11.18.207</ip>
        </controller>
        <controller>
            <ip>10.11.18.208</ip>
        </controller>
        <controller>
            <ip>10.11.18.209</ip>
        </controller>
    </web-config>
    <common>
        <snmp>
             <trap-ip>10.1.1.85</trap-ip>
            <snmp-read-community>gdcbhv</snmp-read-community>
```

```
                <snmp-write-community>nkppui</snmp-write-community>
                <snmp-trap-community>nkppui</snmp-trap-community>
            </snmp>
            <service>
                <protocols>
                    <enable-bittorent>1</enable-bittorent>
                    <enable-edk>1</enable-edk>
                    <enable-http>1</enable-http>
                    <enable-ares>1</enable-ares>
                    <enable_cache_out_port>1</enable_cache_out_port>
                </protocols>
                <bandwidth-management>
                    <bandwidth_exchange_time>2</bandwidth_exchange_time>
                    <bandwidth_calc_time>1</bandwidth_calc_time>
                </bandwidth-management>
                <net>
                    <fwd-mode>BOUNCING</fwd-mode>

                    <bounce id="0">
                    </bounce>
                    <subnet_range_per_link name="a">
                        <cidr_range>10.0.0.0/8</cidr_range>
                    </subnet_range_per_link>
                </net>
                <io>

<selective_cache_in_io_latency_high_wm>20</selective_cache_in_io_latency_high_wm>

<selective_cache_in_io_latency_low_wm>5</selective_cache_in_io_latency_low_wm>
                        <io_latency_high_wm>20</io_latency_high_wm>
                        <io_latency_low_wm>5</io_latency_low_wm>
                </io>
            </service>
        </common>
        <blades>16</blades>
        <blade id="1">
            <cache-engine>
                <network>
                    <network_interfaces number="8">
                        <nic nic_index="0">
                            <name>eth5</name> <!-- 201 -->
                            <nic_detail>IFF_PF_PACKET</nic_detail>
                            <vip>10.138.201.1</vip>
                        </nic>
                        <nic nic_index="1">
                            <name>eth6</name> <!-- 202 -->
                            <nic_detail>IFF_PF_PACKET</nic_detail>
                            <vip>10.138.202.1</vip>
                        </nic>
                         <nic nic_index="2">
                            <name>eth7</name> <!-- 203 -->
                            <nic_detail>IFF_PF_PACKET</nic_detail>
                            <vip>10.138.203.1</vip>
                        </nic>
                        <nic nic_index="3">
                            <name>eth8</name> <!-- 204 -->
                            <nic_detail>IFF_PF_PACKET</nic_detail>
                            <vip>10.138.204.1</vip>
                        </nic>
                        <nic nic_index="4">
                            <name>eth9</name> <!-- 301 -->
                            <nic_detail>IFF_PF_PACKET</nic_detail>
                        </nic>
                        <nic nic_index="5">
```

```
                    <name>eth10</name> <!-- 302 -->
                    <nic_detail>IFF_PF_PACKET</nic_detail>
                </nic>
                <nic nic_index="6">
                    <name>eth11</name> <!-- 303 -->
                    <nic_detail>IFF_PF_PACKET</nic_detail>
                </nic>
                <nic nic_index="7">
                    <name>eth12</name> <!-- 304 -->
                    <nic_detail>IFF_PF_PACKET</nic_detail>
                </nic>
            </network_interfaces>
        </network>
        <service>
            <net>
            <cacheout_bypass_details name="first" nic_index="4">
                <bypass_netmask>255.255.255.0</bypass_netmask>
                <bypass_remote_server_ip1>10.138.31.254</bypass_remote_server_ip1>
                <associated_network_element_index>0</associated_network_element_index>
                <bypass_local_ip>10.138.31.1</bypass_local_ip>
            </cacheout_bypass_details>
            <cacheout_bypass_details name="second" nic_index="5">
                <bypass_netmask>255.255.255.0</bypass_netmask>
                <bypass_remote_server_ip1>10.138.32.254</bypass_remote_server_ip1>
                <associated_network_element_index>1</associated_network_element_index>
                <bypass_local_ip>10.138.32.1</bypass_local_ip>
            </cacheout_bypass_details>
            <cacheout_bypass_details name="first" nic_index="6">
                <bypass_netmask>255.255.255.0</bypass_netmask>
                <bypass_remote_server_ip1>10.138.33.254</bypass_remote_server_ip1>
                <associated_network_element_index>2</associated_network_element_index>
                <bypass_local_ip>10.138.33.1</bypass_local_ip>
            </cacheout_bypass_details>
            <cacheout_bypass_details name="second" nic_index="7">
                <bypass_netmask>255.255.255.0</bypass_netmask>
                <bypass_remote_server_ip1>10.138.34.254</bypass_remote_server_ip1>
                <associated_network_element_index>3</associated_network_element_index>
                <bypass_local_ip>10.138.34.1</bypass_local_ip>
            </cacheout_bypass_details>
        </net>
        </service>
    </cache-engine>
</blade>
...
```


(**Output omitted**)

exit

To exit any mode or close an active CLI session use the **exit** command. When you execute this command in Configuration mode, you are returned to the Enable mode.

exit

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode, Enable mode, Configuration, and Server mode

Usage Guidelines

When you execute this command in Configuration mode, if you have created any configuration change but did not use the **apply** command to implement the changes, the following warning message appears:

```
Exiting configuration mode without apply, will discard changes.
   Are you sure? [N/y] n
```

Examples

The following example shows how to exit Configuration mode:

```
Configuration# exit
```

export

To export the current configuration to a TFTP server, use the **export** command.

export *tftp_server filename*

Syntax Description

tftp_server	The IP address or the hostname of the TFTP server to export the configuration to.
filename	The filename to create when exporting the current configuration file.

Command Modes

Configuration mode

Examples

The following example exports the current configuration to the TFTP server with an IP address of 192.168.0.97:

```
configuration# export 192.168.0.97 current-config
```

help

To display the CLI commands that are available in the current mode, with a short description of each command, use the **help** command.

help

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode, Enable mode, Configuration, and Server mode

Examples

The following example displays the list of CLI commands that are available in Configuration mode:

```
configuration# ?
```

```
apply                Apply config changes
diff                 Show pending changes
discard              Discard pending changes
display              Display current configuration
exit                 Exit current mode
export               Export cluster configuration to TFTP server
help                 Commands description
import               Import cluster configuration from TFTP server
network              Configure management network interface
restore              Restore last good configuration
time                 Set system date and time
```

import

To import the current configuration from a TFTP server, use the **import** command.

import *tftp_server filename*

Syntax Description

tftp_server	The IP address or the hostname of the TFTP server from which to import the configuration. **Note:** You can use **localhost** for the *tftp_server* parameter. If you use **localhost**, then the file must be located in the /tftpboot folder of the VDS TC platform to which you are importing.
filename	The filename that contains the configuration that you want to import.

Command Modes

Configuration mode

Examples

The following example imports a new configuration to the VDS TC platform from the TFTP server 192.168.0.97 and uses the **diff** command to show the differences in the imported configuration file:

```
configuration# import 192.168.0.97 current-config
configuration# diff
  <?xml version="1.0" encoding="UTF-8" standalone="no" ?>

  <cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='cluster_conf.xsd'>
    <mgmt-config>
      <ipaddr>192.168.0.97</ipaddr>
      <netmask>255.255.255.0</netmask>
-     <default-gw>192.168.0.1</default-gw>
+     <default-gw>192.168.0.2</default-gw>
      <nameserver>194.90.1.5</nameserver>
      <alert-email>support@cisco.com</alert-email>
      <site_name>UB1K office - promisc </site_name>
    </mgmt-config>
    <web-config></web-config>
```

network

To change the configuration default gateway or the configuration management network IP address, use the **network** command.

network {**default6_gw** *dg_ipv6_address* | **default_gw** *dg_ipv4_address* | **ip** *ipv4_address netmask* | **ip6** *ipv6_address/ipv6_prefix*}

Note Changing the ip address or default gateway could be dangerous, because if the change is done from a Telnet or SSH session to a remote VDS-TC server, it might end the connection with which you are currently working.

Syntax Description

default6_gw *dg_ipv6_address*	Sets the default IPv6 gateway of the VDS TC platform to the address specified with the *dg_ip_address* parameter.
default_gw *dg_ip_address*	Sets the default gateway of the VDS TC platform to the address specified with the *dg_ip_address* parameter.
ip *ip_address netmask*	Changes the management network interface IP address to the address and subnet mask specified with the provided parameters.
ip6 *ip_v6address/ipv6_prefix*	Configures the management network interface IPv6 address and prefix.

Command Modes

Configuration mode

Examples

The following example sets the default gateway for the VDS TC platform to 192.168.0.2 and sets the management network IP address to 192.168.0.97:

```
configuration# network default_gw 192.168.0.2
configuration# network ip 192.168.0.97 255.255.255.0
```

restore

To restore the previous configuration, use the **restore** command. You must still use the **apply** command to apply the previous configuration.

restore

Syntax Description

This command has no arguments or keywords.

Command Modes

Configuration mode

Examples

The following example restores the previous configuration. Note that restoring the configuration still requires the use of the **apply** command to implement the previous:

```
configuration# restore
configuration# diff
  <?xml version="1.0" encoding="UTF-8" standalone="no" ?>
  <cluster xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='cluster_conf.xsd'>
    <mgmt-config>
      <ipaddr>192.168.0.202</ipaddr>
      <netmask>255.255.255.0</netmask>
      <default-gw>192.168.0.2</default-gw>
      <nameserver>10.1.1.235</nameserver>
      <alert-email>support@cisco.com</alert-email>
      <site_name>192.168.0.202 </site_name>
      <external_syslog_ip>127.0.0.1</external_syslog_ip>
    </mgmt-config>
    <web-config></web-config>
.
.
configuration# apply
applying configuration...
Configuration applied
```

time

To change the system time, use the **time** command.

time *MMDDYYhhmm*

Syntax Description

MMDDYYhhmm	Specifies the new date and time to set on the system.

Command Modes

Configuration mode

Examples

The following example sets the date and time to March 26, 2014 10:27 am:

```
configuration# time 0326141027
Mon March 26 10:27:00 NZST 2014
```

Server Mode Commands

This section describes the commands that are available in Server mode. The Server Mode Commands table lists the commands that are described in this section.

Table 9-4 Server Mode Commands

Command	Description
arp_server	Shows the server arp table.
direction_server	Calculates visible subnets on an interface.
dmesg_server	Displays dmesg.
dstat_server	Displays I/O, CPU, and networking statistics.
exit	Exits current mode.
fdisk_server	Displays available caching block devices.
help	Lists available commands and their descriptions.
ifconfig_server	Displays interfaces.
iostat_server	Displays I/O statistics.
jumbo_server	Echoing Jumbo packets.
lock	Locks the server in Out of Service mode. The server remains locked even when the system is rebooted, until an **unlock** command is used to bring it back to In Service mode.
powercycle	Gracefully shuts down server.
powerdown	Shutdown server
powerup	Boot up server
process_server	Displays process status for the caching application, spread, and apache.
restart	Restarts the server VDS TC application, without rebooting the server.
start	Starts the server VDS TC application, without rebooting the server.
stop	Stops the server VDS TC application, without rebooting the server.
systemid_server	Shows the chassis ID.
unlock	Unlocks the server from Out of State mode and returns it to In Service mode.

arp_server

To display the arp table of the server, use the **arp_server** command.

arp_server

Syntax Description

This command has no arguments or keywords.

Command Modes

Server mode

Examples

The following example shows output from the **arp_server** command:

```
oper server 1# arp_server
Address                  HWtype  HWaddress           Flags Mask            Iface
10.11.14.103             ether   00:80:e5:21:5e:69   C                     eth4
10.11.14.108             ether   00:80:e5:21:37:19   C                     eth4
10.11.14.102             ether   00:80:e5:21:55:79   C                     eth4
ce-2                     ether   00:25:b5:00:01:0f   C                     eth0
mg-1                     ether   44:03:a7:4a:ba:0c   C                     eth0
10.11.14.109             ether   00:80:e5:21:54:09   C                     eth4
10.11.14.100             ether   00:80:e5:21:7f:b1   C                     eth4
10.11.14.105             ether   00:80:e5:21:70:d1   C                     eth4
10.11.14.104             ether   00:80:e5:21:54:a9   C                     eth4
10.11.14.107             ether   00:80:e5:21:56:79   C                     eth4
10.11.14.101             ether   00:80:e5:21:60:61   C                     eth4
10.11.14.106             ether   00:80:e5:21:58:59   C                     eth4
```

direction_server

To calculate the seen subnets on an interface of a server, use the **direction_server** command.

direction_server *interface_name*

Syntax Description

interface_name	The name of the interface for which you want to see the seen subnets.

Command Modes

Server mode

Examples

The following shows output from the **direction_server** command for eth0:

```
oper server 1# direction_server eth0
    358 00:1d:09:6d:3e:4d
     55 00:22:19:5a:e5:1d
    587 00:22:19:5a:f5:08
```

dmsg_server

To display the dmsg of a server, use the **dmsg_server** command.

dmsg_server

Syntax Description

This command has no arguments or keywords.

Command Modes

Server mode

Examples

The following example shows output from the **dmsg_server** command:

```
oper server 1# dmesg_server
enic 0000:13:00.0: eth8: Link UP
[       config 21  718]: Received notification NETDEV_UP for device eth8
[       config  0  718]: Received notification NETDEV_CHANGE for device eth8
enic 0000:14:00.0: eth9: Link UP
[       config 21  718]: Received notification NETDEV_UP for device eth9
[       config  0  718]: Received notification NETDEV_CHANGE for device eth9
enic 0000:15:00.0: eth10: Link UP
[       config 21  718]: Received notification NETDEV_UP for device eth10
[       config  0  718]: Received notification NETDEV_CHANGE for device eth10
enic 0000:16:00.0: eth11: Link UP
[       config 17  718]: Received notification NETDEV_UP for device eth11
[       config  0  718]: Received notification NETDEV_CHANGE for device eth11
enic 0000:17:00.0: eth12: Link UP
[       config 22  718]: Received notification NETDEV_UP for device eth12
[       config  0  718]: Received notification NETDEV_CHANGE for device eth12
[       config  0 1573]: Asked to remove all domain ranges
[       config  0 1524]: Asked to add domain range 10.0.0.0/8
[       config  0 1338]: base 10.0.0.0 mask ffff:ffff:ffff:ffff:ffff:ffff:ff00:0
 bits 104
device eth1 entered promiscuous mode
device eth1 left promiscuous mode
device eth1 entered promiscuous mode
device eth1 left promiscuous mode
```

dstat_server

To display I/O, CPU, and networking statistics for the server, use the **dstat_server** command.

dstat_server [-**N** {*int_name* | **total**}] [*count*]

Syntax Description

-**N** {*int_name* | **total**}	Displays the sent and received statistics for the interface specified with the *int_name* parameter. If you do not specify this parameter, the output displays the total sent and received statistics across all interfaces.
count	The *count* parameter determines the number of seconds between reports. If you do not specify this parameter, the default is 15 seconds.

Note To stop the output from the dstat_server command, press **Ctrl-C**.

Command Modes

Server mode

Examples

The following shows output from the **dstat_server** command for eth0:

```
oper server 2# dstat_server
----total-cpu-usage---- -dsk/total- -net/total- ---paging-- ---system--
usr sys idl wai hiq siq| read  writ| recv  send|  in   out | int   csw
 14  11  61   6   2   5| 101M   12M|   0     0 |   0     0 | 184k  221k
 18  22  48   6   2   5|  77M   25M|  92M  102M|   0     0 | 204k  276k
 19  21  47   6   2   5|  74M   27M|  93M  105M|   0     0 | 205k  263k
```

```
 19  21  48   6   2   5|  81M   25M|  90M  101M|   0     0 | 197k  285k
 15  13  56   8   3   5|  85M   27M|  98M  107M|   0     0 | 193k  237k
 13  13  60   7   2   5|  64M   29M|  88M   99M|   0     0 | 417k  188k
 10  17  64   4   2   4|  36M   11M|  76M   80M|   0     0 | 827k  138k
 14  14  58   6   3   6|  72M   32M|  97M  111M|   0     0 | 502k  198k
 11  17  61   3   2   5|  40M   15M|  88M   89M|   0     0 | 808k  153k
 14  15  58   5   2   5|  69M   18M|  96M  103M|   0     0 | 588k  199k
 13  11  62   5   3   5|  83M   17M| 100M  108M|   0     0 | 288k  203k
 14  12  61   6   3   5|  82M   22M| 103M  112M|   0     0 | 293k  212koper
```

exit

To exit any mode or close an active CLI session use the **exit** command. When you execute this command in Server mode, you are returned to the Enable mode.

exit

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode, Enable mode, Configuration, and Server mode

Examples

The following example shows how to exit Configuration mode:

```
oper server 2# exit
console#
```

fdisk_server

To display the available caching block devices, use the **fdisk_server** command.

fdisk_server

Syntax Description

This command has no arguments or keywords.

Command Modes

Server mode

help

To display the CLI commands that are available in the current mode, with a short description of each command, use the **help** command.

help

Syntax Description

This command has no arguments or keywords.

Command Modes Regular mode, Enable mode, Configuration, and Server mode

Examples The following example displays the list of CLI commands that are available in Server mode:

```
oper server 2# help
arp_server           Show server's arp table
direction_server     Calculate seen subnets on interface
dmesg_server         Display dmesg
dstat_server         Display IO/CPU/Networking  statistics
exit                 Exit current mode
fdisk_server         Display available for caching block devices
help                 Commands description
ifconfig_server      Display interface(s)
iostat_server        Display IO statistics
jumbo_server         Echoing Jumbo packets
lock                 Lock service on server
powercycle           Reboot server
powerdown            Shutdown server
powerup              Boot up server
process_server       Display process status for pang, spread, apache
restart              Server restart
start                Server start
stop                 Server stop
systemid_server      Show chassis id
unlock               Unlock service on server
```

ifconfig_server

To display information about the interfaces on the server, use the **ifconfig_server** command.

ifconfig_server

Syntax Description This command has no arguments or keywords.

Command Modes Server mode

Examples The following example displays the list of CLI commands that are available in Configuration mode:

```
oper server 2# ifconfig_server
bond0     Link encap:Ethernet  HWaddr A4:93:4C:AA:46:E8
          inet addr:10.11.12.2  Bcast:10.11.12.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:6814200598 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2005679242 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2921007474225 (2785689.8 Mb)  TX bytes:242126131117 (230909.4 Mb)

bond1     Link encap:Ethernet  HWaddr D4:8C:B5:BD:14:1E
          inet addr:10.11.14.1  Bcast:10.11.14.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:9000  Metric:1
          RX packets:64603113486 errors:89 dropped:0 overruns:0 frame:89
          TX packets:36953489594 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
```

```
          RX bytes:252394952746728 (240702584.0 Mb)  TX bytes:13368105910600 (12748819.2
Mb)

bond1:1   Link encap:Ethernet  HWaddr D4:8C:B5:BD:14:1E
          inet addr:10.11.15.1  Bcast:10.11.15.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:9000  Metric:1

bond1:2   Link encap:Ethernet  HWaddr D4:8C:B5:BD:14:1E
          inet addr:10.11.16.1  Bcast:10.11.16.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:9000  Metric:1

bond1:3   Link encap:Ethernet  HWaddr D4:8C:B5:BD:14:1E
          inet addr:10.11.17.1  Bcast:10.11.17.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:9000  Metric:1

eth0      Link encap:Ethernet  HWaddr A4:93:4C:AA:46:E8
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:4244115125 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2005679242 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1489663253279 (1420653.5 Mb)  TX bytes:242126131117 (230909.4 Mb)

eth1      Link encap:Ethernet  HWaddr A4:93:4C:AA:46:E8
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:2570085472 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1431344220224 (1365036.2 Mb)  TX bytes:0 (0.0 b)

eth2      Link encap:Ethernet  HWaddr D4:8C:B5:BD:14:1C
          UP BROADCAST RUNNING MULTICAST  MTU:2020  Metric:1
          RX packets:42265364 errors:0 dropped:0 overruns:0 frame:8
          TX packets:179411738864 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:16384
          RX bytes:2746295632 (2619.0 Mb)  TX bytes:256030627850416 (244169833.9 Mb)

eth2.301  Link encap:Ethernet  HWaddr D4:8C:B5:BD:14:1C
          UP BROADCAST RUNNING PROMISC MULTICAST  MTU:2020  Metric:1
          RX packets:4965310 errors:0 dropped:0 overruns:0 frame:0
          TX packets:44851936447 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:16384
          RX bytes:211957172 (202.1 Mb)  TX bytes:63647415701475 (60698905.6 Mb)
<output omitted>
```

iostat_server

To display I/O statistics for the server, use the **iostat_server** command.

iostat_server [-**t** *interval*] [-**k** *count*]

Syntax Description

-t *interval*	The amount of time in seconds between each report. The default is 5 seconds.
-k *count*	Used in conjunction with the interval parameter. If the *count* parameter is specified, the count determines the number of reports generated at the specified interval. If the interval parameter is specified without the *count* parameter, the iostat command generates reports continuously, until you press **Ctrl-C**.

Command Modes

Server mode

Examples

The following shows output from the **iostat_server** command for eth0:

```
oper server 2# iostat_server -k 1
Linux 2.6.27.7-llpf-9-default (ce-2)    08/18/2009      _x86_64_

Time: 02:59:55 PM
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          14.39    0.00   25.56    6.20    0.00   61.08

Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz
await  svctm  %util
sda               2.45    20.02    0.23    3.52     4.21    95.23    53.08     0.09
23.02   0.52   0.19
sda1              0.00     0.00    0.00    0.00     0.01     0.00    56.26     0.00
7.11   6.67   0.00
sda2              1.84     3.17    0.10    1.19     2.39    17.32    30.47     0.00
0.69   0.37   0.05
sda3              0.12     0.22    0.03    0.26     0.68     2.21    19.66     0.00
1.10   0.75   0.02
sda4              0.48    16.63    0.09    2.06     1.13    75.70    71.28     0.09
39.35   0.70   0.15
sdb               0.03     0.00   15.02    1.48   891.64    94.32   119.56     0.27
16.43  12.70  20.95
sdb1              0.01     0.00    0.00    0.00     0.01     0.00    56.43     0.00
13.57  12.29   0.00
<output omitted>
```

jumbo_server

To send jumbo echo messages, use the **jumbo_server** command.

jumbo [**-c** *counter*] [**-I** {*IP_address* | *interface*}] *destination*

Syntax Description

-c *counter*	The number of times the request is generated.
-I {*IP_address* | *interface*}	The interface IP address or interface name from which the echo requests are sent.
destination	The destination to which the jumbo echo message will be sent.

Command Modes Server mode

Examples The following example sends jumbo echo messages to 192.168.5.117 sourced from the eth0 interface:

```
oper server 1# jumbo_server -I eth0 192.168.5.117
PING 192.168.5.117 (192.168.5.117) from 192.168.5.117 eth0: 8972(9000) bytes of data.
8980 bytes from 192.168.5.117: icmp_seq=1 ttl=64 time=0.043 ms
8980 bytes from 192.168.5.117: icmp_seq=2 ttl=64 time=0.024 ms
8980 bytes from 192.168.5.117: icmp_seq=3 ttl=64 time=0.028 ms
8980 bytes from 192.168.5.117: icmp_seq=4 ttl=64 time=0.033 ms
8980 bytes from 192.168.5.117: icmp_seq=5 ttl=64 time=0.039 ms

--- 192.168.5.117 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3998ms
rtt min/avg/max/mdev = 0.024/0.033/0.043/0.008 ms
```

lock

To lock the server in Out of Service mode, use the **lock** command. After you enter this command, the system will ask you to confirm the command.

lock

Syntax Description This command has no arguments or keywords.

Command Modes Server mode

Usage Guidelines The server remains locked even after a system reboot. To bring the server back to In Service mode, use the **unlock** command.

Examples The following is an example of using the **lock** command:

```
oper server 2# lock
Are you sure (y/n)? y
Locking server...
Locked
oper server 2# exit
console# show status
Cluster state: degraded

Blade Slot   Status   Operational state   Device state    Administrative state
ce-1          N/A            N/A            stopped              N/A
ce-2         powered on     disabled        started             locked
ce-3         powered on     enabled         started            unlocked
ce-4         powered on     enabled         starting           unlocked
```

powercycle

To gracefully reboot the server, use the **powercycle** command. After you enter this command, the system will ask you to confirm the command.

powercycle

Syntax Description

This command has no arguments or keywords.

Command Modes

Server mode

Examples

The following is an example of using the **powercycle** command:

```
oper server 2# powercycle
Are you sure (y/n)? y
OK
oper server 2# exit
console# show status
Cluster state: degraded

Blade Slot   Status   Operational state   Device state    Administrative state
ce-1          N/A            N/A            stopped              N/A
ce-2          N/A            N/A            stopped              N/A
ce-3        powered on     enabled          started            unlocked
ce-4        powered on     enabled          started            unlocked
```

process_server

To display the process status for the caching application, spread, and apache, use the **process_server** command.

process_server

Syntax Description

This command has no arguments or keywords.

Command Modes

Server mode

Examples

The following is output from the **process_server** command:

```
oper server 2# process_server
spread    7194     1  9 Aug17 ?        01:54:55 /usr/bin/spread -n ce-2 -c
/etc/spread.conf
root      7195     1  2 Aug17 ?        00:34:11 /opt/pang/cache/avalon/sbin/snmpd -f -A
-LF e /opt/pang/cache/avalon/var/log/snmpd.log -LS c u 10.11.12.3
root      7769     1 99 Aug17 ?        3-21:03:05 /opt/pang/bin/pang -d -f
/opt/pang/conf/pang.conf
admin    10336 10335 44 15:02 ?        00:00:00 bash -c /usr/bin/sudo
/opt/pang/bin/check_processes.sh 2> /dev/null
root     10366 10336  0 15:02 ?        00:00:00 sh /opt/pang/bin/check_processes.sh
```

powerdown

To shutdown the server, use the **powerdown** command. After you enter this command, the system will ask you to ocnfirm the command.

powerdown

Syntax Description

This command has no arguments or keywords.

Command Modes

Server mode

powerup

To boot up the server, use the **powerup** command. After you enter this command, the system will ask you to ocnfirm the command.

powerup

Syntax Description

This command has no arguments or keywords.

Command Modes

Server mode

restart

To restart the server VDS TC application, use the **restart** command. After you enter this command, the system will ask you to confirm the command.

restart

Syntax Description

This command has no arguments or keywords.

Command Modes

Server mode

Examples

The following is an example of using the **restart** command:

```
oper server 2# restart
Are you sure (y/n)? y
Restarting server 2
oper server 2# exit
console# show status
Cluster state: enabled

Server Slot   Status   Operational state   Device state   Administrative state ce-1
powered on       enabled       started              unlocked
```

```
ce-2         powered on       enabled       starting             unlocked
ce-3         powered on       enabled       started              unlocked
ce-4         powered on       enabled       started              unlocked
ce-5         powered on       enabled       started              unlocked
ce-6         powered on       enabled       started              unlocked
ce-7         powered on       enabled       started              unlocked
ce-8         powered on       enabled       started              unlocked
ce-9         powered on       enabled       started              unlocked
ce-10        powered on       enabled       started              unlocked
ce-11        powered on       enabled       started              unlocked
ce-12        powered on       enabled       started              unlocked
```

start

To start the server VDS TC application, use the **start** command. After you enter this command, the system will ask you to confirm the command.

start

Syntax Description

This command has no arguments or keywords.

Command Modes

Server mode

Examples

The following is an example of using the **start** command:

```
oper server 2# start
Starting server 2
oper server 2# exit
console# show status
Cluster state: enabled

Blade Slot   Status   Operational state   Device state   Administrative state ce-1
powered on       enabled       started              unlocked
ce-2         powered on       N/A           N/A                  unlocked
ce-3         powered on       enabled       started              unlocked
ce-4         powered on       enabled       started              unlocked
ce-5         powered on       enabled       started              unlocked
ce-6         powered on       enabled       started              unlocked
ce-7         powered on       enabled       started              unlocked
ce-8         powered on       enabled       started              unlocked
ce-9         powered on       enabled       started              unlocked
ce-10        powered on       enabled       started              unlocked
ce-11        powered on       enabled       started              unlocked
ce-12        powered on       enabled       started              unlocked
```

stop

To stop the server VDS TC application, use the **stop** command. After you enter this command, the system will ask you to confirm the command.

stop

Syntax Description This command has no arguments or keywords.

Command Modes Server mode

Examples The following is an example of using the **stop** command:

```
oper server 2# stop
Are you sure (y/n)? y
Stopping server 2
```

systemid_server

To display the chassis ID of the server, use the **systemid_server** command.

systemid_server

Syntax Description This command has no arguments or keywords.

Command Modes Server mode

Examples The following output from the **systemid_server** command:

```
oper server 1# systemid_server
FCH1623V2QJ
```

unlock

To unlock the server state from Out of Service mode and return it to In-Service mode, use the **unlock** command. Note that the **unlock** command causes VDS TC to go through a stop and start cycle.

unlock

Syntax Description This command has no arguments or keywords.

Command Modes Server mode

Examples The following is an example of using the **unlock** command:

```
console# oper server 2
oper server 2# unlock
Unlocking server...
Unlocked
```

Monitoring VDS TC (Cluster)

You can view statistical information regarding bandwidth utilization, caching statistics, and server status for VDS-TC using both SNMP and the CLI. This chapter describes the CLI commands that you can use to monitor VDS TC.

show config

To view the current active configuration in XML format, use the **show config** command.

show config

Syntax Description This command has no arguments or keywords.

Command Modes Enable mode

show connectivity

To view the iSCSI connectivity, use the **show connectivity** command.

show connectivity

Syntax Description This command has no arguments or keywords.

Command Modes Regular mode and Enable mode

Examples The following example shows output from the **show connectivity** command.

```
console# show connectivity
Blade   Active iSCSI sessions
ce-1    9
ce-2    9
ce-3    9
ce-4    9
```

```
ce-5    9
ce-6    9
ce-7    9
ce-8    9
ce-9    9
ce-10   9
ce-11   9
ce-12   9
ce-13   9
ce-14   9
ce-15   9
ce-16   9
Do you want to see more detailed information ? [y|n]  :y
Please enter blade number [1-16] :3
10.11.14.101
10.11.14.104
10.11.14.105
10.11.14.102
10.11.14.103
10.11.14.106
10.11.14.107
10.11.14.109
10.11.14.108
```

show eth_status

To view the eth status, use the **show eth_status** command.

show eth_status

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following example shows output from the **show eth_status** command.

```
console> show eth_status
Blade     eth0    eth1    eth2    eth3    eth4    eth5    eth6    eth7    eth8    eth9
eth10   eth11   eth12
ce-1       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-2       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-3       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-4       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-5       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-6       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-7       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-8       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
```

```
ce-9       UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-10      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-11      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-12      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-13      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-14      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-15      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
ce-16      UP      UP      UP      UP      UP      UP      UP      UP      UP      UP
UP      UP      UP
```

show eventlog

To display the content of the event log, use the **show eventlog** command.

show eventlog

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **show eventlog** command:

```
console> show eventlog
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol9         mounted          active    ce-1
678        546        131        19.37
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol10        mounted          active    ce-1
678        548        129        19.15
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol11        mounted          active    ce-1
678        546        132        19.51
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol12        mounted          active    ce-1
678        546        131        19.44
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol13        mounted          active    ce-1
678        546        132        19.49
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol14        mounted          active    ce-1
678        544        134        19.76
Apr 22 09:57:11 ce-1 pang[29533]: /mnt/vol15        mounted          active    ce-1
678        543        135        19.94
Apr 22 10:06:59 ce-1 pang[29533]: volume             state      availability owner
total      free       used       usage
<output omitted>
```

show leader

To display the hostname of the current cluster leader, use the **show leader** command.

show leader

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode or Enable mode

Examples

The following is sample output from the **show eventlog** command:

```
console> show leader
ce-1
```

show license

To view the current active license, use the **show license** command.

show license

Syntax Description

This command has no arguments or keywords.

Command Modes

Enable mode

Examples

The following example shows output from the **show license** command.

```
console# show license
Licensed chassis serial number: DGB9C4J
EDK enabled:                            1
Bittorent enabled:                      1
Kazaa enabled:                          1
Gnutella enabled:                       1
Ares enabled:                           1
Http enabled:                           1
Pando enabled:                          1
Thunder enabled:                        0
Smartfilter enabled:                    0
Netflix enabled:                        1
Silverlight enabled:                    1
Storage volumes:                        120
Controllers:                            1
CDR logs:                               1
Service Detection:                      1
web Cache enabled:                      0
N_PLUS_K enabled:                       0
Evaluation ends on:                     14-5-2014
Max bandwidth:                          unlimited
Max forwarding: 48000 Mbps
```

show process

To display the status of VDS TC components (caching application, spread, and apache) as they run on the platform, use the **show process** command.

show process

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following example shows output from the **show process** command.

```
console> show process
wwwrun    1954     1  0 Feb27 ?        01:00:30 python ubview.py --logfile
/opt/pang/mgmt/django/run/ubview.log --pidfile /opt/pang/mgmt/django/run/ubview.pid
--reactor epoll
root      2709     1  0 Feb19 ?        00:00:43 /opt/pang/mgmt//avalon/sbin/snmptrapd -f
-Osq -Ls u -c /opt/pang/mgmt//avalon/sbin/snmptrapd.conf 10.11.12.1
wwwrun    3309  6877  0 22:03 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun    3310  6877  0 22:03 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
ntp       6669     1  0 Feb25 ?        00:10:29 /usr/sbin/ntpd -p /var/run/ntp/ntpd.pid -n
-g -u ntp:ntp -U 0 -i /var/lib/ntp -c /etc/ntp.conf
admin     6690  6689  0 Feb19 pts/1    00:00:00 pang_cli
root      6877     1  0 Feb25 ?        00:01:32 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun    8225  6877  0 22:05 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
admin    11721 11718  0 21:55 pts/9    00:00:00 -pang_cli
wwwrun   12317  6877  0 21:55 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun   12490  6877  0 22:06 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
root     14479     1  2 Feb19 ?        19:55:00 /opt/pang/mgmt/bin/monitor
wwwrun   14861  6877  0 22:07 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun   14862  6877  0 22:07 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
spread   16390     1  7 Feb24 ?        2-14:24:48 /usr/bin/spread -n mg-1 -c
/etc/spread.conf
wwwrun   18636  6877  0 21:57 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
root     21280     1  0 Feb27 ?        03:26:14 /opt/pang/mgmt/avalon/sbin/snmpd -f -A -LF
e /opt/pang/mgmt/avalon/var/log/snmpd.log -LS c u 0.0.0.0
wwwrun   24200  6877  0 21:49 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
wwwrun   24321  6877  0 21:49 ?        00:00:00 /usr/sbin/httpd2-prefork -f
/etc/apache2/httpd.conf
```

show status

To view the per server service status, physical slot status, administration status (enabled/disabled) and operational status, use the **show status** command.

show status

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **show status** command:

```
Console> show status
Cluster state: enabled

Server Slot          Status               Operational state    Device state
Administrative state
ce-1                 powered on           enabled              started
unlocked
ce-2                 powered on           enabled              started
unlocked
ce-3                 powered on           enabled              started
unlocked
ce-4                 powered on           enabled              started
unlocked
ce-5                 powered on           enabled              started
unlocked
ce-6                 powered on           enabled              started
unlocked
ce-7                 powered on           enabled              started
unlocked
ce-8                 powered on           enabled              started
unlocked
ce-9                 powered on           enabled              started
unlocked
ce-10                powered on           enabled              started
unlocked
ce-11                powered on           enabled              started
unlocked
ce-12                powered on           enabled              started
unlocked
ce-13                powered on           enabled              started
unlocked
ce-14                powered on           enabled              started
unlocked
ce-15                powered on           enabled              started
unlocked
ce-16                powered on           enabled              started
unlocked
```

show systemid

To view the unique system identifier, which is used for support and licensing purposes, use the **show systemid** command.

show systemid

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following is sample output from the **show systemid** command:

```
console> show systemid
console# show systemid
*FCH1652V1FJ
FCH16337ULN
FCH16337UL6
FCH16337U58
FCH16337UEY
FCH1651J08C
FCH16287KMP
FCH16327FCB
FCH16337TXB
FCH1651J0KM
FCH16337E0P
FCH16337UH2
FCH1651J0KH
FCH1651J0E9
FCH1651J0Y7
FCH16487F10
FCH16487ES1
```

show time

To view the system date and time, use the **show time** command.

show time

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following example shows output from the **show time** command.

```
console> show time
Tue Apr 01 2014 00:37:03 GMT+0200
```

show uptime

To view the system up time of individual servers, use the **show uptime** command.

show uptime

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following example shows output from the **show uptime** command.

```
console> show uptime
cluster              4 days, 7h:34m:09s
ce-1                 4 days, 7h:34m:39s
ce-2                 4 days, 7h:34m:38s
ce-3                 4 days, 7h:34m:36s
ce-4                 4 days, 7h:34m:34s
ce-5                 4 days, 7h:34m:34s
ce-6                 4 days, 7h:34m:36s
ce-7                 4 days, 7h:34m:32s
ce-8                 4 days, 7h:34m:30s
ce-9                 4 days, 7h:34m:29s
ce-10                4 days, 7h:34m:28s
ce-11                4 days, 7h:34m:27s
ce-12                4 days, 7h:34m:26s
ce-13                4 days, 7h:34m:28s
ce-14                4 days, 7h:34m:23s
ce-15                4 days, 7h:34m:22s
ce-16                4 days, 7h:34m:21s
```

show version

To view the installed software version for management server and per caching engine server, use the **show version** command.

show version

Syntax Description

This command has no arguments or keywords.

Command Modes

Regular mode and Enable mode

Examples

The following example shows output from the **show version** command.

```
console> show version
VDS-TC Transparent Caching           cli version 5.7.3b54
management           VDS-TC Transparent Caching mgmt software version 5.7.3b54
ce-1                 5.7.3b54            LLPF Version LLPF_05.7.3b53-54
```

```
ce-2                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-3                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-4                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-5                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-6                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-7                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-8                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-9                5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-10               5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-11               5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-12               5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-13               5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-14               5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-15               5.7.3b54            LLPF Version LLPF_05.7.3b53-54
ce-16               5.7.3b54            LLPF Version LLPF_05.7.3b53-54
```

CLI Reference (Cluster)

The following is a tree structure of the CLI commands that are available in a VDS-TC Cluster configuration.

Regular Mode

```
arp                     Display ARP table
current_cli_users       Show currently logged in cli users
direction               Calculate the visible subnets on the interface
dmesg                   Display the dmesg
dstat                   Display hardware, software, and I/O statistics
enable                  Enter Enable mode
eventlog                Event log commands
    date                Display eventlog of a specific date
    export              Export event log to TFTP server
    show                Display event log
    tail                Display online event log
exit                    Logs you out from the CLI
help                    Display CLI commands
ifconfig                Display the interface(s)
iostat                  Display extended I/O statistics
jumbo                   Send jumbo echo messages
ping                    Send echo messages
show                    Show run-time information
    cluster-bus-ip      Display cluster communication bus status
    connectivity        Display the iSCSI connectivity
    detection_rules     Display detection rules
    dstat               Display IO statistics
    eth_status          Display the eth status
    eventlog            Display event log
    leader              Display current cluster leader
    process             Display status of VDS TC components
    status              Display cluster administrative and application status
    systemid            Display system serial number
    time                Display system date and time
    uptime              Display cluster uptime
    version             Display software version
    volumes             Display mounted volumes
tcpdump                 Dump traffic on a network interface
traceroute              Display a packet's route
```

Enable Mode

Enable mode includes the commands available in Regular mode, and commands available only in Enable mode, as follows:

```
access                    Manage system access parameters
    enable-password       Enable mode password
    idle-session-timeout  Set idle session timeout value
    user-password         Regular mode password
apache_restart            Restart apache
arp                       Display ARP table
cache                     Cache operations
    black_list            Manage the cache black list.
        add               Add a file to the black list using hash ID and protocol.
        dump              Display (dump) the entire black list.
        export            Export the black list to a TFTP server.
        remove            Remove a file from the black list using a hash ID and protocol.
    hash                  Display the file's metadata using a hash ID.
    list                  Display and exports the list of cache content.
        display           Display full list of cache content.
        export            Export cache content to TFTP server.
        short             Display the Least Recently Used cached HASH IDs.
    remove                Remove a file from the cache using hash ID.
    summary               Display CMDB statistics summary.
    sync                  Verify and synchronize cache metadata
    volume                Manage cache volumes
        activate          Activate a cache volume
        deactivate        Stop using a specific volume for caching
        insert            Add new cache volume
        remove            Remove all hash IDs from a specific volume
        remove_content    Erase volume's metadataf
config                    Enter Configuration mode
current_cli_users         Show currently logged in cli users
detection_rules           Manage detection rules configuration
    apply                 Apply detection rules
    export_groups         Export groups to a TFTP server
    export_signatures     Export signatures configuration to TFTP server
    import_groups         Import groups from TFTP server
    import_signatures     Import signatures configuration from TFTP server
    show                  Display current detection rules
direction                 Calculate the visible subnets on the interface
dmesg                     Display the dmesg
dstat                     Display I/O statistics
eventlog                  Event log commands
    date                  Display eventlog of a specific date
    export                Export event log to TFTP server
    forward               Starts event log forwarding
    show                  Display event log
    stop                  Stops event log forwarding
    tail                  Display online event log
exit                      Logs you out from the CLI
help                      Display CLI commands
ifconfig                  Display the interface(s)
iostat                    Display extended I/O statistics
jumbo                     Send jumbo echo messages
license                   Manage system license
    activate              Activate system license
    import                Import license from TFTP server
    show                  Display current license
oper                      System management operations
    server                Server operations
    service               Manage services
        start             Start VDS TC software and services
```

```
        stop                Stop VDS TC software and all its services
ping                        Send echo messages
reset                       Reset management services
show                        Show run-time information
    cluster-bus-ip          Display cluster communication bus status
    config                  Display running configuration
    connectivity            Display the iSCSI connectivity
    detection_rules    Display detection rules
    dstat              Display IO statistics
    eth_status              Show the status of the ethernet interfaces
    eventlog                Display event log
    leader                  Display current cluster leader
    license                 Display system license information
    process                 Display status of VDS TCVDS TC components
    status                  Display cluster administrative and application status
    systemid                Display system serial number
    time                    Display system date and time
    uptime                  Display cluster uptime
    version                 Display software version
    volumes                 Display mounted volumes
system                      System information
    logs export             Export the system logs to TFTP server
    statistics export       Export the system statistics to TFTP server
traceroute                  Display a packet's route
upgrade                     Upgrade VDS TC software version
    all                     Upgrade VDS TC software on all servers
    management              Upgrade VDS TC software for management server
    server                  Upgrade VDS TC software for specific server
vlan                        Display a list of users currently logged in
    add                     Add a vlan to an interface
    remove                  Remove a vlan from an interface
```

Configuration Mode

```
apply          Apply config changes
cluster-bus-ip Configure cluster bus type
diff           Show pending changes
discard        Discard pending changes
display        Display pending configuration
exit           Exit current mode
export         Export cluster configuration to TFTP server
help           Commands description
import         Import cluster configuration from TFTP server
network        Configure management network interface
    default6_gwConfigure IPv6 default gateway
    default_gw Configure IPv4 default gateway
    ip         Configure IPv4 management network interface
    ip6        Configure IPv6 management network interface
restore        Restore last good configuration
time           Set system date and time
```

Server Mode

```
arp_server         Display server's ARP table
direction_server   Calculate visible subnets on interface
dmesg_server       Display dmesg
dstat_server       Display I/O, CPU, and networking statistics
exit               Enter Enabled mode
```

```
fdisk_server       Display available caching block devices
help               Display available commands
ifconfig_server    Display interface(s)
iostat_server      Display I/O statistics
jumbo_server       Echo jumbo packets
lock               Lock server in out-of-service mode
powercycle         Graceful server shutdown
powerdown          Service power down
powerup            Service power up
process_server     Display process status for pang, spread, apache
restart            Restart server VDS TC application
start              Start server VDS TC application
stop               Stop server VDS TC application
systemid_server    Show chassis ID
unlock             Unlock server from out-of-state to in-service mode
```