



Cisco TelePresence Video Communication Server

ADMINISTRATOR GUIDE

Software version X5.1
November 2010



Introduction Overview and status System configuration Cisco VCS configuration Zones and neighbors Clustering and peers Call processing Bandwidth control Firewall traversal Applications Maintenance Appendices

What's in this manual?

CISCO TELEPRESENCE VIDEO COMMUNICATION SERVER
ADMINISTRATOR GUIDE

Introduction	
The Cisco TelePresence Video Communication Server	8
Overview	8
Cisco VCS and the video communication network	8
Cisco VCS base applications	9
Cisco VCS Control	9
Cisco VCS Expressway™	9
Standard features	10
Optional features	11
FindMe™	11
Device Provisioning	11
Dual Network Interfaces	11
What's new in this version?	12
Usability enhancements	12
FindMe™ / User Policy option key	12
Subzone registration policies	12
Zone configuration	12
Conference Factory generated alias ranges	12
Cisco AM GW support	12
Far end camera control interworking	13
G.729 support for interworked calls	13
Advanced account security	13
CRL checking for TLS connections to LDAP servers	13
HTTPS client certificate validation	13
Clustering	13
Hardware failure warnings	13
Auditor account access level	13
Remote account authentication over LDAP	13
Starter Pack	13
Using this guide	14
Using the Cisco VCS	15
Web interface	15
Installation and initial configuration	15
Using the web interface	15
Supported browsers	15
Page features and layout	16
Command line interface (CLI)	17
Installation and initial configuration	17
Command types	17
How CLI commands are shown in this guide	17
Text entry	17
Supported characters	17
Case sensitivity	17
Overview and status	
Overview page	19
Status	20
System information	20
Ethernet status	20
IP status	21
Resource usage	21
Registrations by device	22
Registrations by alias	22
Registration history	23
Registration details	23
Calls	24
Call history	24
Call summary	24
Searches	25
Search history	25
Search details	25
Local Zone	26
Zones	26
Links	27
Pipes	27
TURN relays	28
Presence	28
OCS Relay	29
System configuration	
System configuration	41
System administration	41
About the system name	41
About administration access settings	41
Ethernet	41
About Ethernet speed	41
IP	42
About IP protocols	42
IPv4 to IPv6 gatewaying (interworking)	42
External LAN interface	42
About IP routes (static routes)	42
About LAN configuration	42
About Dual Network Interfaces	42
About static NAT	42
Quality of Service (QoS)	43
Supported mechanisms	43
DNS	43
About DNS servers	43
About DNS settings	43
Time	44
About the NTP server	44
VCS time display	44
Login page	44
SNMP	45
About SNMP	45
External manager	45
About external managers	45

Introduction

Overview and
status

System
configuration

Cisco VCS
configuration

Zones and
neighbors

Clustering and
peers

Call
processing

Bandwidth
control

Firewall
traversal

Applications

Maintenance

Appendices

What's in this manual?

CISCO TELEPRESENCE VIDEO COMMUNICATION SERVER
ADMINISTRATOR GUIDE

Configuration	45	SIP protocols and ports.....	51	Zones and neighbors	
Logging	46	Session timers	51	Introduction.....	62
About Event Log levels	46	SIP device interoperability	51	Local Zone and subzones.....	63
Setting the Event Log level	46	SIP domains	51	Configuring the Local Zone and its subzones.....	63
About remote logging.....	46	Interworking	52	Bandwidth management.....	63
Enabling remote logging.....	46	Configuring interworking.....	52	Local Zone searches.....	63
Cisco VCS configuration		H.323 <-> SIP interworking mode	52	Traversal Subzone.....	64
H.323	48	Enabling SIP endpoints to dial H.323 numbers.....	52	What are traversal calls?.....	64
H.323 overview	48	Registration control	53	Configuring the Traversal Subzone ports	64
About H.323 on the VCS	48	Registration overview.....	53	Zones	65
Using the Cisco VCS as an H.323 gatekeeper	48	Endpoint registration.....	53	About zones	65
H.323 endpoint registration	48	Registrations on a Cisco VCS Expressway.....	53	Neighbor zone	65
Configuring H.323.....	49	MCU, gateway and Content Server registration	53	Traversal client zone	65
Enabling H.323.....	49	Finding a Cisco VCS with which to register	54	Traversal server zone	65
Configuring H.323 ports	49	SIP	54	ENUM zone.....	66
Registration conflict mode.....	49	H.323.....	54	DNS zone	66
Time to live.....	49	Preventing automatic registrations.....	54	Default Zone.....	66
Call time to live.....	49	Device authentication	55	Zone configuration	67
Auto discover.....	49	Device authentication using LDAP.....	56	TLS certificate verification of neighbor systems	67
Caller ID	49	Configuring the LDAP server directory	56	Connections to neighbor systems over TCP and TLS ...	67
SIP	50	Configuring LDAP server settings	56	SIP authentication trust	67
SIP overview	50	Authentication using a local database	57	Configuring zones	68
About SIP on the Cisco VCS.....	50	Authenticating with external systems.....	57	Configuring neighbor zones	68
Using the Cisco VCS as a SIP registrar.....	50	Registering aliases	58	Configuring traversal client zones	69
Using the Cisco VCS as a SIP Presence Server.....	50	About alias registration	58	Configuring traversal server zones	70
Using the Cisco VCS as a SIP proxy server	50	Attempts to register using an existing alias	58	Configuring ENUM zones	71
Proxying registration requests	50	Blocking registrations	58	Configuring DNS zones.....	71
SIP endpoint registration.....	50	Allow and Deny Lists	59	Zone configuration: advanced settings.....	72
Movi v2.0 (or later) clients	50	About Allow and Deny Lists	59	Zone configuration: pre-configured profile settings.....	75
Configuring SIP	51	Activating use of Allow or Deny Lists	59	Dial plans.....	76
Enabling SIP	51	Removing existing registrations	59	Structuring your dial plan	76
SIP registration expiry	51	Using the Allow and Deny Lists	60	Flat dial plan.....	76
SIP registration proxy mode.....	51			Structured dial plan	76
				Hierarchical dial plan	76

Introduction

Overview and
status

System
configuration

Cisco VCS
configuration

Zones and
neighbors

Clustering and
peers

Call
processing

Bandwidth
control

Firewall
traversal

Applications

Maintenance

Appendices

What's in this manual?

CISCO TELEPRESENCE VIDEO COMMUNICATION SERVER
ADMINISTRATOR GUIDE

Clustering and peers

Clustering overview.....	78
About clustering.....	78
About the configuration master.....	78
Cluster configuration.....	79
Configuring clusters.....	79
Setting up a cluster.....	79
Maintaining a cluster.....	79
Setting configuration for the cluster.....	79
Adding and removing peers from a cluster.....	79
Cluster name.....	79
Changing the master peer.....	79
Monitoring the status of the cluster.....	79
Which configuration is not replicated?.....	80
Troubleshooting cluster replication problems.....	80
Managing clusters and peers.....	81
Sharing registrations across peers.....	81
Sharing bandwidth across peers.....	81
Upgrades and downgrades.....	81
Backup and restore.....	81
Clustering and FindMe.....	82
Clustering and Presence.....	82
Clustering and Cisco TMS.....	82
Cluster Subzone.....	83
Neighboring the local Cisco VCS to another VCS cluster ...	83

Call processing

Introduction.....	85
Search process.....	85
Dialing by address types.....	86
About the different address types.....	86
Dialing by IP address.....	86
Endpoints registered to a Cisco VCS Expressway.....	86
Dialing by H.323 ID or E.164 alias.....	86

Dialing by H.323 or SIP URI.....	86
Dialing by ENUM.....	86
Hop counts.....	87
About hop counts.....	87
Configuring hop counts.....	87
Searches and transforms.....	88
Overview of searches and transforms.....	88
Zone searching and transform process.....	88
Pre-search transforms.....	89
Pre-search transform process.....	89
Configuring pre-search transforms.....	90
Search configuration.....	91
Calls to unknown IP addresses.....	91
About unregistered endpoints.....	91
Calls to Unknown IP addresses settings.....	91
Recommended configuration for firewall traversal.....	91
Fallback alias.....	91
Zone searching and zone transforms.....	92
Examples.....	94
Stripping @domain for dialing to H.323 numbers.....	94
Examples.....	95
Transforms for alphanumeric H.323 ID dial strings.....	95
Combining match types and priorities.....	96
Always query a zone with original alias (no transforms).....	96
Configuring a zone for incoming calls only.....	96
Filter queries to a zone without transforming.....	97
Allowing calls to IP addresses only if they come from known zones.....	97
Query a zone for original and transformed alias.....	98
Query a zone for two or more transformed aliases.....	99
Call Policy.....	100
Call Policy and authentication.....	100
Enabling Call Policy.....	101
Configuring basic Call Policy using the web interface.....	101

Configuring Call Policy using a CPL script.....	102
Viewing existing CPL script.....	102
About CPL XSD files.....	102
Uploading a CPL script.....	102
Deleting an existing CPL script.....	102
Routing calls via the Cisco AM GW.....	103
VCS and the Cisco AM GW.....	103
Configuring the VCS to use the Cisco AM GW.....	103
Usage features and limitations.....	103
Controlling which calls to route via the Cisco AM GW.....	103
Configuring Cisco AM GW policy rules.....	104
URI dialing.....	105
Overview.....	105
URI dialing without DNS.....	105
URI dialing via DNS.....	105
Enabling URI dialing via DNS.....	105
URI resolution process using DNS.....	106
URI dialing via DNS for outgoing calls.....	107
URI dialing process.....	107
Adding and configuring DNS zones.....	107
Configuring search rules for DNS zones.....	107
Configuring DNS servers.....	107
URI dialing via DNS for incoming calls.....	108
Types of DNS records required.....	108
Incoming call process.....	108
SRV record format.....	108
Configuring H.323 SRV records.....	108
Configuring SIP SRV records.....	108
Example DNS record configuration.....	109
URI dialing and firewall traversal.....	109
Recommended configuration.....	109
ENUM dialing.....	110
ENUM dialing process.....	110
Enabling ENUM dialing.....	110

Introduction

Overview and
status

System
configuration

Cisco VCS
configuration

Zones and
neighbors

Clustering and
peers

Call
processing

Bandwidth
control

Firewall
traversal

Applications

Maintenance

Appendices

What's in this manual?

ENUM dialing for outgoing calls	111	Links.....	122	Configuring the Cisco VCS as a traversal client.....	135
Prerequisites	111	Creating and editing links.....	122	Configuring the Cisco VCS as a traversal server	136
Process.....	111	Default links	122	Adding and configuring a traversal server zone	136
ENUM dialing for outgoing calls	112	Pipes	123	Configuring traversal for endpoints	136
Adding and configuring ENUM zones	112	Creating and editing pipes.....	123	Configuring traversal server ports.....	137
Configuring matches for ENUM zones.....	112	Applying pipes to links	124	Configuring the Cisco VCS as a TURN server.....	138
Configuring transforms for ENUM zones	112	Default bandwidth and downspeeding.....	125	TURN services.....	138
Configuring DNS servers	112	Bandwidth control examples	126	About ICE	138
ENUM dialing for incoming calls.....	113	Example without a firewall.....	126	About TURN.....	138
Prerequisites	113	Example with a firewall.....	127	TURN relay server	138
About DNS domains for ENUM.....	113	Cisco VCS Expressway subzone configuration.....	127	Capabilities and limitations	138
Configuring DNS NAPTR records	113	Cisco VCS Control subzone configuration	127	Configuring TURN services	138
Call configuration.....	114	Firewall traversal		Applications	
Call routed mode	114	Firewall traversal overview.....	129	Conference Factory.....	140
Call loop detection mode	114	About Expressway™.....	129	Configuration	140
Call IDs, Serial Numbers and Tags.....	115	Cisco VCS as a firewall traversal client.....	129	Presence.....	141
Identifying calls	115	Cisco VCS as a firewall traversal server	129	Presence Server	141
Disconnecting calls.....	116	Quick guide to traversal client - server configuration	130	Presence User Agent (PUA).....	142
Limitations when disconnecting SIP calls	116	Firewall traversal protocols and ports	131	Aggregation of presence information	142
Bandwidth control		Expressway process	131	FindMe presence	142
Bandwidth control overview	118	H.323 firewall traversal protocols	131	Registration refresh period.....	142
Bandwidth control on the VCS	118	SIP firewall traversal protocols.....	131	Configuring Presence	143
Example network deployment.....	118	Ports for initial connections from traversal clients	132	Enabling and disabling Presence Services.....	143
Subzones	119	Assent ports.....	132	Presence Server expiration times	143
About subzones	119	SIP ports.....	132	Viewing presence status	144
Subzone registration policy.....	119	H.460.18/19 ports.....	132	Publishers	144
About the Traversal Subzone	119	TURN ports.....	132	Presentities	144
Traversal calls	119	Ports for connections out to the public internet.....	132	Subscribers.....	144
About the Default Subzone.....	119	Firewall traversal and authentication	133	OCS Relay	145
Default links between subzones	119	Authentication and NTP.....	133	Configuring OCS Relay.....	145
Configuring subzones and membership rules.....	120	Other issues.....	134	Viewing OCS Relay status.....	145
Applying bandwidth limitations to subzones	121	Firewall traversal and Dual Network Interfaces.....	134	FindMe™.....	146
Bandwidth consumption of traversal calls	121	Firewall configuration	134	Overview	146

What's in this manual?

What is FindMe?	146	Upgrading and option keys	155	TLS encryption and CRL checking	165
How are devices specified?	146	Installing and rebooting.....	155	Root account.....	166
Process overview	146	Upgrade procedure	156	Changing the root account password	166
Who must do what before FindMe can be used?.....	146	Upgrading using secure copy (SCP/PSCP)	156	Accessing the root account over SSH and Telnet	166
Recommendations when deploying FindMe	146	Downgrade procedure	156	Resetting passwords	166
Enabling FindMe on the Cisco VCS	147	Option keys	157	System administration access.....	167
Configuring FindMe	147	Adding option keys using the web interface.....	157	Overview	167
Caller ID	147	Adding option keys using the CLI	157	Security considerations.....	167
Device creation message	147	Security certificates.....	158	Configuring administration access	167
Searching for FindMe accounts	147	Enabling security	158	Administration session timeout	167
Principal devices.....	147	Trusted CA certificate	158	Backup and restore	168
FindMe™ user guide	148	Server certificate data.....	158	Creating a backup of your VCS data	168
About FindMe	148	Advanced account security.....	159	Restoring a previous backup	168
FindMe user accounts.....	148	Prerequisites	159	System snapshot.....	169
Individual versus group FindMe	148	VCS functionality: changes and limitations.....	159	Creating a system snapshot	169
Accessing the FindMe home page	148	Enabling advanced account security	159	Incident reporting	170
Configuring your FindMe user account.....	149	Login accounts	160	Warning: privacy-protected personal data.....	170
FindMe home page	149	Account authentication configuration	160	What information does the report contain?	170
Defining device details	150	Administrator accounts	161	Viewing incident reports	171
Defining location details.....	151	Default administrator account	161	Sending incident reports manually.....	171
Provisioning (Starter Pack)	152	Additional administrator accounts.....	161	Removing sensitive information from the report	171
Configuration	152	Administrator password security.....	161	Sending incident reports automatically	171
TMS Agent.....	153	Maintaining administrator accounts	161	Tools.....	172
Overview	153	Maintaining user accounts	162	Check pattern.....	172
FindMe™	153	Creating user accounts	162	Locate.....	172
Device Provisioning.....	153	Managing user accounts	163	Port usage.....	173
TMS Agent account passwords	153	Configuring devices and locations	163	Local VCS inbound ports	173
Maintenance		Configuring principal devices	163	Local VCS outbound ports	173
Upgrading software components	155	Changing an account's password.....	163	Remote listening ports	173
Overview	155	Administrator and user groups.....	164	Restarting, rebooting and shutting down.....	174
VCS software components	155	Configuring administrator groups	164	Restoring default configuration.....	175
Prerequisites	155	Configuring user groups.....	164	Configuration items reset by DefaultValuesSet level 3	175
Backing up before upgrading	155	Account authentication using LDAP.....	165	Configuration items reset by DefaultValuesSet level 2	176
		Configuring LDAP server settings	165	Password encryption.....	178

What's in this manual?

Appendices

CPL reference.....	180
Overview of CPL on the Cisco VCS	180
address-switch	180
address	180
field.....	181
subfield.....	182
otherwise	182
not-present.....	182
location.....	183
rule-switch.....	183
proxy	184
reject	184
Unsupported CPL elements.....	184
CPL examples.....	185
Call screening of authenticated users	185
Call screening based on alias.....	185
Call screening based on domain	186
Change of domain name.....	186
Allow calls from locally registered endpoints only	187
Block calls from Default Zone and Default Subzone..	187
Restricting access to a local gateway.....	188
Using the address-switch node.....	188
Using the taa:rule-switch node.....	188
Redirecting failed calls based on status code.....	189
Reject attempts to subscribe to a presentity	190
Regular expression reference	191
Pattern variable reference	192
Port reference	193
DNS configuration.....	196
Overview	196
Verifying the SRV record.....	196
Microsoft DNS server.....	196
BIND 8 & 9	196

LDAP configuration for device authentication.....	197
About the LDAP databases.....	197
Downloading the H.350 schemas	197
Microsoft Active Directory	197
Prerequisites	197
Installing the H.350 schemas.....	197
Adding H.350 objects	198
Securing with TLS	198
OpenLDAP	199
Prerequisites	199
Installing the H.350 schemas	199
Adding H.350 objects	200
Securing with TLS	200
Command reference - xConfiguration	201
Command reference - xCommand.....	248
Command reference - xStatus	265
Warnings	284
Bibliography	287
Glossary.....	289
Legal notices.....	295

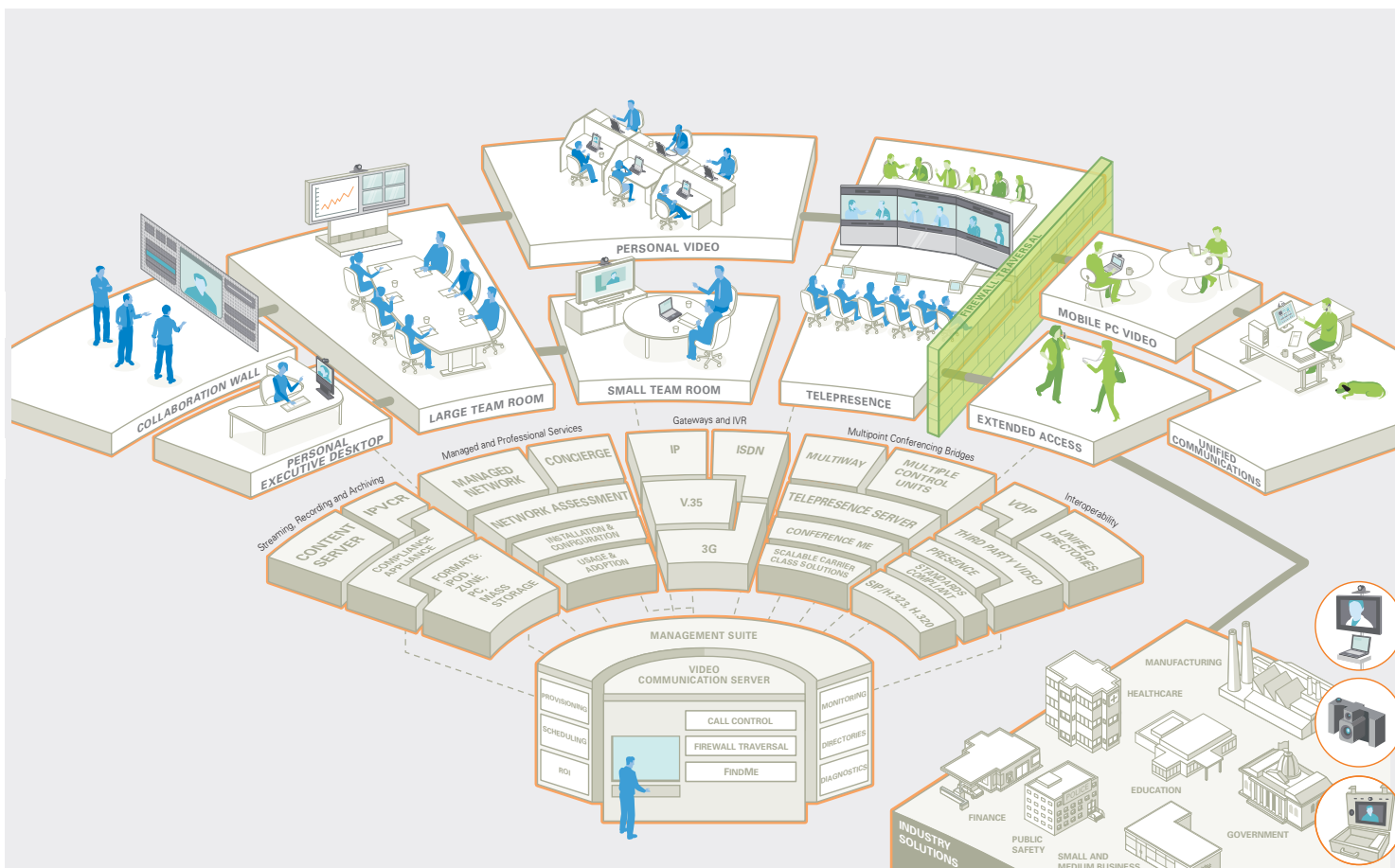
The Cisco TelePresence Video Communication Server

CISCO TELEPRESENCE VIDEO COMMUNICATION SERVER
ADMINISTRATOR GUIDE

Overview

The Cisco TelePresence Video Communication Server (Cisco VCS) enhances the video experience and provides seamless communication between SIP and H.323 devices utilizing IETF and ITU standards. The Cisco VCS is the center of the video communication network, and connects all H.323 and SIP endpoints, infrastructure, and management devices. The Cisco VCS provides unrivaled scalability and redundancy to video communications, and is integral to Cisco interoperability with unified communications and Voice over IP systems. The Cisco VCS can be deployed with either the [Control](#) application or the [Expressway™](#) application, with various optional packages including [FindMe™](#), [Dual Network Interfaces](#) and [Device Provisioning](#).

Cisco VCS and the video communication network



Introduction

Overview and
status

System
configuration

Cisco VCS
configuration

Zones and
neighbors

Clustering and
peers

Call
processing

Bandwidth
control

Firewall
traversal

Applications

Maintenance

Appendices

The Cisco TelePresence Video Communication Server

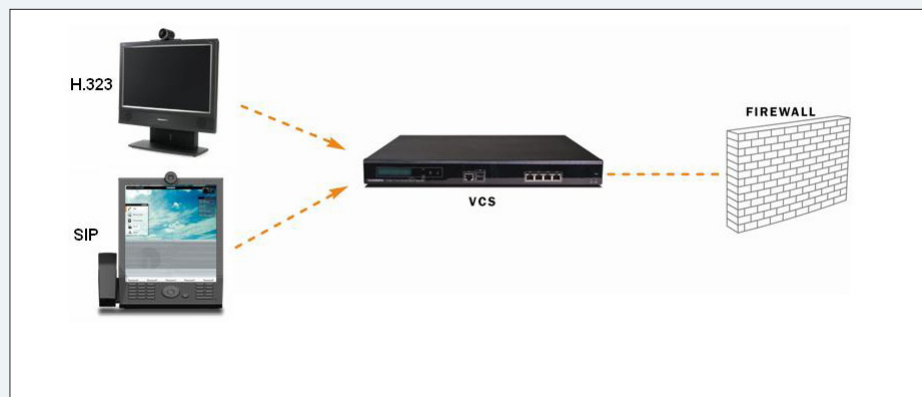
CISCO TELEPRESENCE VIDEO COMMUNICATION SERVER
ADMINISTRATOR GUIDE

Cisco VCS base applications

Cisco VCS Control

The Cisco VCS Control provides internal video control and administration for all SIP and H.323 devices. It is normally deployed within your wide area network with endpoints that are behind the same firewalls or NAT devices.

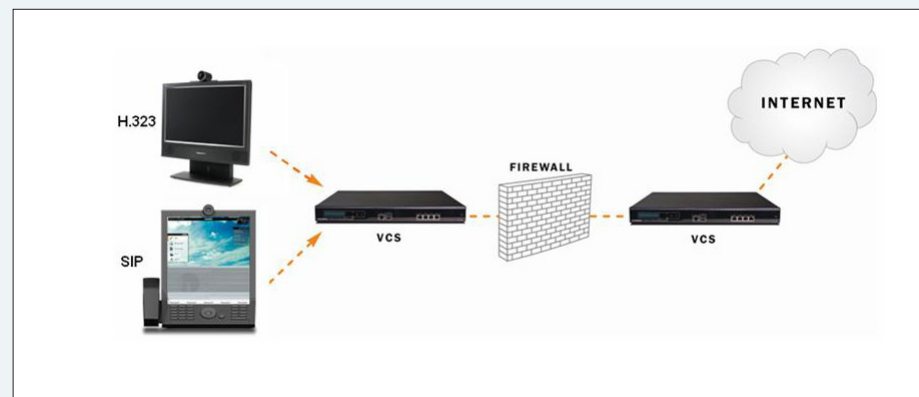
The Cisco VCS Control replaces the need to have separate H.323 gatekeeper, SIP registrar and H.323 - SIP gateway servers.



Cisco VCS Expressway™

The Cisco VCS Expressway provides standards-based firewall traversal for SIP and H.323 devices allowing secure firewall traversal of any firewall or NAT device. As well as all the functionality of a Cisco VCS Control, it also provides registration of traversal-enabled devices and can act as a standards-based TURN server.

The Cisco VCS Expressway is normally deployed outside of your firewall or within the DMZ.



Standard features

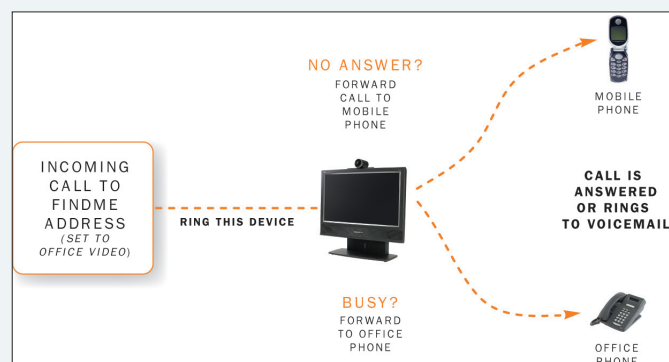
- 2500 endpoint registrations
- H.323 gatekeeper
- SIP Proxy/Registrar
- SIP Presence Server
- SIP Presence User Agent
- SIP and H.323 support, including SIP/H.323 gatewaying
- IPv4 and IPv6 support, including IPv4/IPv6 gatewaying
- QoS tagging
- Bandwidth management on both a per-call and a total usage basis, configurable separately for calls within the local subzones and to external systems and zones
- Automatic downspeeding option for calls that exceed the available bandwidth
- URI and ENUM dialing via DNS, enabling global connectivity
- Up to 500 non-traversal calls
- Up to 100 traversal calls
- 1000 external zones with up to 2000 matches
- 1000 subzones and supporting up to 3000 membership rules
- Flexible zone configuration with prefix, suffix and regex support
- Can function as a standalone VCS or be neighbored with other systems such as VCSs, Border Controllers, gatekeepers and SIP proxies
- n+1 redundancy, can be part of a cluster of up to 6 VCSs for increased capacity and redundancy
- Intelligent Route Director for single number dialing and network failover facilities
- Optional endpoint authentication
- Control over which endpoints are allowed to register
- Call Policy (also known as Administrator Policy) including support for CPL
- Can be managed with Cisco TelePresence Management Suite (Cisco TMS) 12.5 or later
- AD authentication for administrators of the VCS
- Pre-configured defaults for Microsoft Office Communications Server (OCS) 2007 neighbor zones
- Pre-configured defaults for Cisco Call Manager neighbor zones
- Pre-configured defaults for Nortel Communication Server neighbor zones
- Embedded setup wizard using a serial port for initial configuration
- System administration using a web interface or RS-232, Telnet, SSH, and HTTPS

Optional features

The following features are available on the VCS by the purchase and installation of the appropriate option key:

FindMe™

A unique industry solution that gives individual video users a single alias on which they can be contacted regardless of location. Users have the ability to log on to a Web-based interface and control where and how they are contacted. The FindMe feature also includes support for Microsoft Office Communications Server (OCS) 2007, enabling FindMe aliases to register as Microsoft Office Communicator (MOC) clients, and MOC clients to view the presence status of FindMe aliases.



Device Provisioning

The **Device Provisioning** option key allows VCS to provision endpoints with configuration information on request and to supply endpoints with phone book information. (Endpoints including Movi v2.0 or later, and E20 v2.1 or later can request to be provisioned.) All configuration and phone book information is managed in Cisco TMS, and distributed to the clients through the TMS Agent running on the VCS. The TMS Agent on the VCS also provides Cisco TMS with the provisioned client's status.

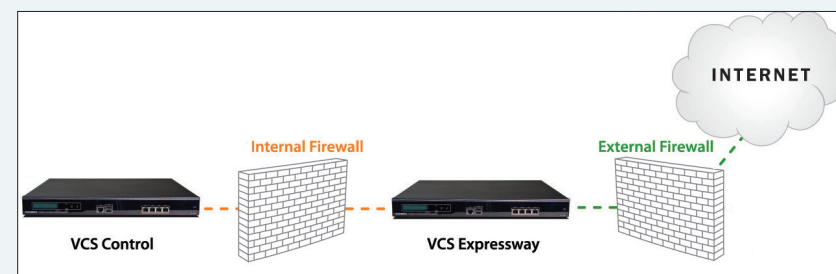
There is no configuration associated with **Device Provisioning** on the VCS – it is either on or off, depending on whether or not the option key is installed. See the Cisco TMS documentation and the [Provisioning Deployment Guide \[26\]](#).

Dual Network Interfaces

Enables the LAN 2 Ethernet port on the VCS Expressway, allowing you to have a secondary IP address for your VCS.

This option also includes support for deployments where a VCS Expressway is located behind a static NAT device, allowing it to have separate public and private IP addresses.

This configuration is intended for high-security deployments where the VCS Expressway is located in a DMZ between two separate firewalls on separate network segments.



What's new in this version?

The following features have been introduced in version X5.1 of the VCS software:

Usability enhancements

Many usability improvements have been made, including:

- **Description fields for configuration items:** a free-format description can be specified for the following configuration items: transforms, Allow List and Deny List patterns, search rules, subzone membership rules.
When viewing the summary list of these items the description is displayed as a mouse-over tooltip.
- **Enable and disable configuration items:** transforms, search rules and subzone membership rules can be individually enabled and disabled. This makes it easier to make or test configuration changes.
Previously, configuration items would have to be deleted and re-created as necessary. The enabled or disabled state is clearly shown on the summary list pages.
- **"Add suffix" and "add prefix" transform options:** new pre-search transform pattern behavior options let you add a prefix or suffix to the matching alias. Previously, regular expressions would have been required to do this.
- **Consistent create and modify behavior for configuration items:** for configuration where multiple items can be defined (for example, Allow List patterns, search rules and so on) the create and modify behavior has been made consistent so that you are always returned to the summary list page after saving your changes.
Additionally, new search rules, subzone membership rules, subzones and zones are all created in a single step. Previously you had to specify some of the configuration values when creating the item, and then return to the edit page to specify the remainder of the values.
- **"Please select" in drop-down fields:** when creating configuration items some of the default values presented in drop-down selection fields have been replaced with a "please select" value. This helps prevent potentially undesirable default values being selected by mistake.

- **Improved filtering options for Event Log and Configuration Log:** advanced filtering options let you include or exclude specific words or phrases when filtering the view of the Event Log and the Configuration Log.
- **OCS Relay status:** colored status icons make the difference between the online and offline OCS Relay status more distinct.
- **Configuration warnings:** more warnings are raised for common misconfiguration scenarios, for example if a clustered VCS has H.323 disabled, or if default links are not present.

FindMe™ / User Policy option key

- The User Policy option key has been renamed as the FindMe™ option key.
- "FindMe accounts" and "FindMe groups" are now referred to as "user accounts" and "user groups" respectively.

Subzone registration policies

- In addition to using Allow Lists and Deny Lists, registrations can be controlled at the subzone level. Each subzone can be configured to allow or deny registrations assigned to it via the subzone membership rules. See the [subzones](#) section for more information.
- Up to 3000 subzone membership rules (previously 2000) can be configured across all subzones.

Zone configuration

- **TLS authentication:** the VCS can perform certificate verification of neighbor zones when communicating over TLS. Mutual authentication can be performed with neighbor VCSs.
- **TLS default transport type:** TLS is the default SIP transport type when setting up new neighbor, traversal client and traversal server zones.
- **SIP authentication trust:** the VCS can be configured to trust incoming SIP messages from specified neighbor zones, rather than challenge them. This applies even if device authentication is enabled on the VCS.

- **Accept proxied registrations:** controls whether proxied SIP registrations routed through a neighbor, traversal client or traversal server zone are accepted.
- **Nortel Communication Server 1000 zone profile:** the VCS can automatically configure the settings required for connections to a Nortel Communication Server 1000.
- **Separate authentication credentials per traversal client zone:** each traversal client zone specifies its own username and password for authentication with the traversal server. This allows a traversal client VCS to connect to one or more service providers.

See the [zone configuration](#) section for more information.

Note that the existing **Outbound connection credentials** username and password are still used for connections to all other (non traversal server) external systems.

Conference Factory generated alias ranges

- The upper and lower range limits of the numeric portion of the generated alias can be specified.
- The numeric portion of the generated alias will pad with leading zeroes to maintain a constant length, for example a range of 10-999 will generate aliases 010 through 999.

Cisco AM GW support

The Cisco TelePresence Advanced Media Gateway (Cisco AM GW) provides support for transcoding between standard codecs (such as H.264) and Microsoft RT Video to allow high definition calls between Microsoft Office Communicator (MOC) clients and Cisco endpoints.

- **Advanced Media Gateway zone profile:** automatically configures the VCS with the zone settings required for connection to an Cisco AM GW.
- **Policy rules:** ability to define policy rules to control whether all or only selected calls to or from MOC clients are diverted through the Cisco AM GW.

See [VCS and the Cisco AM GW](#) for more information.

What's new in this version? (continued)

Far end camera control interworking

The VCS supports far end camera control (FECC) when interworking calls between SIP and H.323 endpoints.

G.729 support for interworked calls

The G.729 codec is supported for interworked calls.

Advanced account security

An [Advanced account security](#) option key enables advanced security features and restrictions for high-security installations. Contact your Cisco support representative for more information about this feature.

CRL checking for TLS connections to LDAP servers

Certificate revocation lists (CRLs) can be uploaded and used to verify certificates presented by an LDAP server to the VCS when forming a TLS connection. See [Configuring LDAP server settings](#) for more information.

HTTPS client certificate validation

- The VCS web server can be configured to request a valid client certificate before establishing an HTTPS session with a client system (typically a web browser).
- Client certificate revocation lists can be uploaded and used to verify the client certificate.

See [Security certificates](#) for more information.

Clustering

- **Improved resilience:** cluster configuration replication is more resilient to network delay.
- **TURN server:** relay allocation requests are redirected to other cluster peers if the first VCS's relays are fully allocated.

Hardware failure warnings

Improved hardware failure detection, warnings and status display.

Auditor account access level

An **Auditor** access level can be assigned to administrator accounts and groups. Users with the Auditor privilege can only access the Overview, Event Log and Configuration Log pages.

Remote account authentication over LDAP

The LDAP server address of the remote directory service can be specified as a DNS SRV record, thus allowing multiple (primary and backup) servers to be specified.

Starter Pack

The **Starter Pack** option key is only available as a pre-configured factory setting. It is designed for single box deployments and provides basic device provisioning for registered Movi users, without the need for Cisco TMS. It supports device authentication and supplies phone book information to provisioned devices.

The **Starter Pack** includes the following features:

- Expressway
- FindMe

It has the following license restrictions:

- 50 registrations
- 5 calls (any combination of traversal and non-traversal calls)

Note that installing additional call license option keys will have no effect while the **Starter Pack** option key is present.

See [Provisioning \(Starter Pack\)](#) for more information.

How to use this Administrator Guide

This Administrator Guide is provided to help you make the best use of your Cisco TelePresence Video Communication Server.

Your approach to this documentation depends on what you want to do and how much you already know.

The Administrator Guide has been divided into several sections, each providing different information. In some places information is duplicated between sections to let you have all the relevant information in one place.

This document does not have an index. This is intentional; if the [Table of contents](#) does not direct you to the information you need, you can use the **Find** function in Adobe Reader to search the text for keywords.

Note that the Administrator Guide describes a fully equipped version of the Cisco VCS. Your version may not have all the described extensions installed.

Our main objective with this Administrator Guide is to address your goals and needs. Please let us know how well we succeeded!

Typographical conventions

Most configuration tasks on the VCS can be performed by using either the web interface or a command line interface (CLI). This guide describes how to use both methods.

Web interface

In this guide, instructions for performing a task using the web interface are shown in the format:

- **Menu > Submenu**

followed by the **Name** of the page that you will be taken to.

Command line interface (CLI)

In this guide, instructions for performing a task using the command line interface (CLI) are shown in the format:

- **xConfiguration <Element> <SubElement>**
- **xCommand <Command>**

These are meant as a reference only. Each command is hyperlinked to the [Command reference](#) table at the back of this guide; clicking on the hyperlink will take you to the appropriate section of the table showing all the available sub-elements, parameters and valuespaces for the given command.

Note that:

- Typing the given **xConfiguration** path into the CLI will return a list of values currently configured for that element (and sub-elements where applicable).
- Typing the given **xConfiguration** path into the CLI followed by a ? will return information about the usage for that element and sub-elements.
- Typing the given **xCommand** command into the CLI with or without a ? will return information about the usage of that command.

Web interface

Installation and initial configuration

Full installation and initial configuration instructions for the Cisco VCS are contained in the [VCS Getting Started Guide \[28\]](#).

Using the web interface

To use the web interface:

1. Open a browser window and in the address bar type either:
 - the IP address of the system
 - the FQDN of the system
2. Click **Administrator Login**.
3. Enter a valid administrator **username** and **password** and click **Login**. (See the [Login accounts](#) section for details on setting up administrator accounts.)

You are presented with the **Overview** page.



When logging in using the VCS web interface, you may receive a warning message regarding the VCS's security certificate. This can safely be ignored.

How page navigation is shown in this guide

In this Administrator Guide, instructions for navigating the web interface are shown in the format:

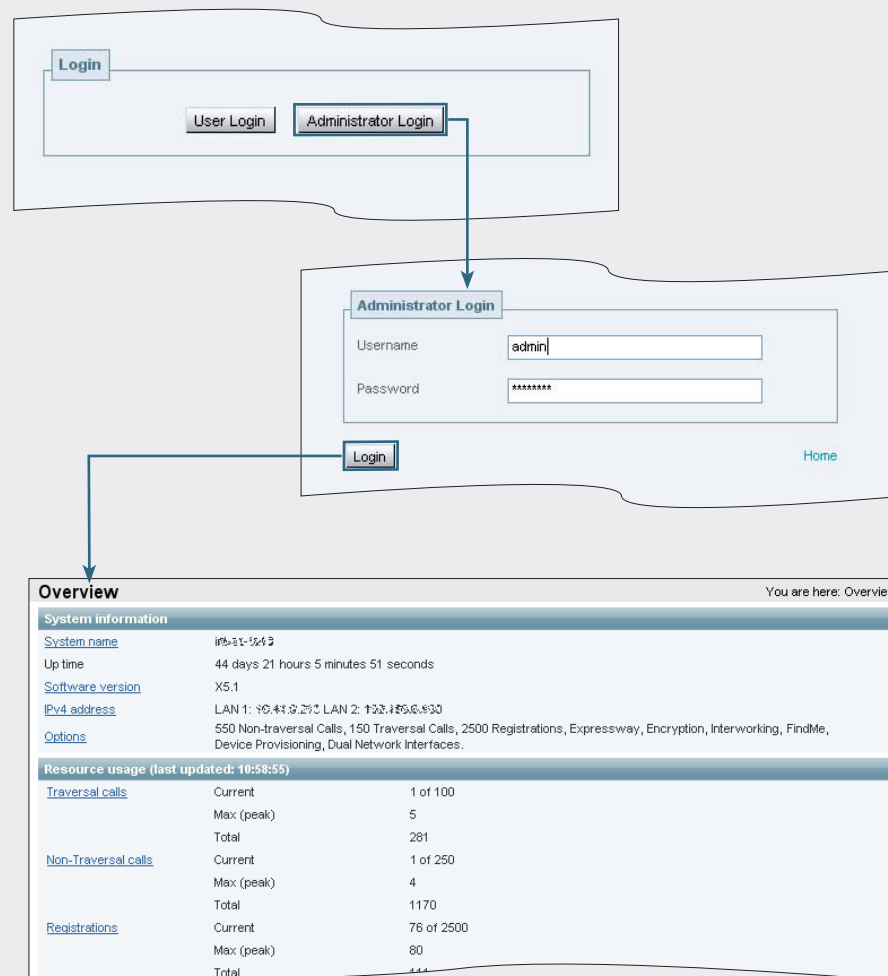
- **Menu option1 > Menu option2**

followed by the **Name** of the page that you will be taken to in order to perform a task.

Supported browsers

The VCS web interface is designed for use with Internet Explorer 6 or later, and Firefox 2 or later. It may work with Opera and Safari, but you could encounter unexpected behavior.

Javascript and cookies must be enabled to use the VCS web interface.



Web interface

Page features and layout

These are the features that can be found on some or all of the web interface pages.

Note that you cannot change configuration settings if your admin account is read-only.

Information bar

The VCS provides you with feedback in certain situations, for example when settings have been saved or when you need to take further action. This feedback is given in a yellow information bar at the top of the page.

Sorting Columns

Click on column headings to sort the information in ascending and descending order.

Select All and Unselect All

Use these buttons to select and unselect all items in the list.

Status

On configuration pages, this section shows you the current status of the items you are configuring.

Note that some configuration changes require a restart to take effect, so if you have changed the configuration but not yet restarted this shows the existing (unchanged) status.

System Information

The name of the user currently logged in, the system name (or LAN 1 IPv4 address if no system name is configured), local system time, hardware serial number and VCS software version are shown at the bottom of the page.

Page name and Location

Every page shows the page name and the menu path to that page. Each part of the menu path is a link; clicking on any of the higher level menu items takes you to that page.



System Warning

This icon appears on the top right corner of every page when there is a system warning in place. Click on this icon to go to the **Warnings** page which gives information about the warning and its suggested resolution.



Help

This icon appears on the top right corner of every page. Clicking on this icon opens a new browser window with help specific to the page you are viewing. It gives an overview of the purpose of the page, and introduces any concepts configured from the page.



Log out

This icon appears on the top right corner of every page. Clicking on this icon ends your administrator session.



Manual

This icon appears on the top right corner of every page. Clicking on this icon takes you directly to the latest version of the Cisco VCS Administrator Guide on our support web site.

Information box

An information box appears on the configuration pages whenever you either click on the **Information** icon or click inside a field. This box gives you information about the particular field, including where applicable the valid ranges and default value.

To close the information box, click on the **X** at its top right corner.



Information

This icon appears to the right of most input fields in the web interface. Clicking on this icon activates the **Information Box**.

The screenshot shows the Cisco VCS web interface. At the top, there's a navigation bar with links: Overview, Status, System configuration, VCS configuration, Applications, Maintenance, Manual, Help, and Logout. Below this is a yellow information bar with a message: "Configuration warning: Restriction policy must be set to 'DenyList' for the Deny List to be active." The main content area is divided into two sections. The top section is titled "Registration Deny List" and contains a table with columns: Pattern, Type, Description, and Actions. The table lists three entries: "@spam.com" (Suffix), "@hacker.net" (Suffix), and "test" (Prefix). Below the table are buttons for "New", "Delete", "Select all", and "Unselect all". The bottom section is titled "Configuration" and contains fields for "NTP server" (158.13.128.33) and "Time zone" (Europe/London). There is a "Save" button and an "Information" box that provides details about the NTP server field, including its range (0 to 128 characters). At the bottom of the page, there's a "Status" section showing the current state (Failed) and reason (No response from NTP server). The footer displays user information (User: admin), system name (Server: 158.13.128.33), system time (09:58 BST), and software version (X5.1).

Command line interface (CLI)

Installation and initial configuration

Full installation and initial configuration instructions for the Cisco VCS are contained in the [VCS Getting Started Guide \[28\]](#).

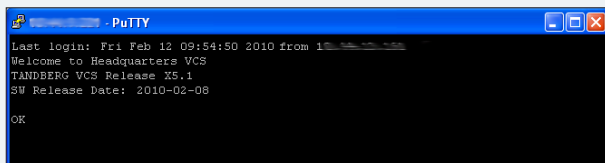
Using the CLI

The command line interface (CLI) is available by default over SSH and through the serial port. Access using Telnet can also be enabled.

To use the CLI:

1. Start a SSH or Telnet session.
2. Enter the IP address or FQDN of the VCS.
3. Login with a username of **admin** and your system password.

You will see a screen similar to that shown below. You can now start using the CLI by typing the appropriate commands.



Command types

Commands are divided into different groups:

xStatus	These commands return information about the current status of the system. Information such as current calls and registrations is available through this command group.
xConfiguration	These commands allow you to add and edit single items of data such as IP address and zones.
xCommand	These commands allow you to add and configure items and obtain information.
xHistory	These commands provide historical information about calls and registrations.

xFeedback

These commands provide information about events as they happen, such as calls and registrations.

See the [Command reference](#) Appendix for a full description of commands available on the VCS.

How CLI commands are shown in this guide

In this guide, instructions for performing a task using the CLI are shown in the format:

- [xConfiguration <Element> <SubElement>](#)
- [xCommand <Command>](#)

These are meant as a reference only. Each command is hyperlinked to the [Command reference](#) table at the back of this guide; clicking on the hyperlink takes you to the appropriate section of the table showing all the available sub-elements, parameters and valuespaces for the given command.

Note that:

- Typing the given **xConfiguration** path into the CLI returns a list of values currently configured for that element (and sub-elements where applicable).
- Typing the given **xConfiguration** path into the CLI followed by a ? returns information about the usage for that element and sub-elements.
- Typing the given **xCommand** command into the CLI with or without a ? returns command usage information.

Text entry

Supported characters

The VCS supports the following characters when entering text in the CLI and web interface:

- the letters **A-Z** and **a-z**
- decimal digits (**0-9**)
- underscore (**_**)
- minus sign / hyphen (**-**)
- equals sign (**=**)
- plus sign (**+**)
- at sign (**@**)
- comma (**,**)
- period/full stop (**.**)
- exclamation mark (**!**)
- spaces
- FindMe account names additionally allow the use of all uppercase and lowercase Unicode characters

The following characters are specifically not allowed:

- tabs
- angle brackets (**<** and **>**)
- ampersand (**&**)
- caret (**^**)

Note that some specific text fields have different restrictions and these are noted in the relevant sections of this guide, including:

- [Administrator and user groups](#)

Case sensitivity

Text items entered through the CLI and web interface are case insensitive.

The only exception is passwords which are case sensitive.

Overview and status

This section describes the information that appears on the **Overview** page and all the pages under the **Status** menu of the web interface.

These pages provide information on the current status and configuration of the Cisco VCS.



Overview page

The **Overview** page summarizes the current configuration and status of your Cisco VCS.

The **Overview** page opens automatically when you first log on to the web interface. You can also access it at any time by clicking on the **Overview** menu at the top left of any page.



Many of the items on this page are configurable, and contain links to the page where they can be configured. For example, clicking on **System name** will take you to the **System administration** page, from where you can configure the system name.

The **Overview** page displays the following information:

System information	
System name	The name that has been assigned to the VCS.
Up time	The amount of time that has elapsed since the system last restarted.
Software version	The version of software that is currently installed on the VCS.
IPv4 address	The VCS's IPv4 address(es).
IPv6 address	The VCS's IPv6 address(es).
Options	The maximum number of calls and registrations, and the availability of additional VCS features such as TURN Relays, FindMe™, Device Provisioning and Dual Network Interfaces, are controlled through the use of option keys . This section shows all the options that are currently installed on the VCS.
Resource Usage	
Traversal calls	Current: The number of traversal calls going through the VCS at this moment. Max (peak): The highest number of concurrent traversal calls handled by the VCS since it was last restarted. Total: The total number of traversal calls handled by the VCS since it was last restarted. See the Traversal calls section for details on what constitutes a traversal call.
Non-traversal calls	Current: The number of non-traversal calls going through the VCS at this moment. Max (peak): The highest number of concurrent non-traversal calls handled by the VCS since it was last restarted. Total: The total number of non-traversal calls handled by the VCS since it was last restarted.
Registrations	Current: The number of endpoints registered to the VCS at this moment. Max (peak): The highest number of endpoints concurrently registered to the VCS since it was last restarted. Total: The total number of registrations on the VCS since it was last restarted.
TURN relays (Cisco VCS Expressway only)	Current: The number of active TURN relays at this moment. Max (peak): The highest number of concurrently active TURN relays since the VCS was last restarted. Total: The total number of TURN relays allocated on the VCS since it was last restarted.

System information

The **System information** page provides details of the software, hardware, and time settings of the VCS.

To view the **System information** page:

- **Status > System > Information**



Some of the items on this page are configurable, and contain links to the page where they can be configured. For example, clicking on **Software options** will take you to the **Option keys** page, from where you can install new optional features.

The page displays the following information:

System information

System name	The name that has been assigned to the VCS.
Product	This will be TANDBERG VCS.
Software version	The version of software that is currently installed on the VCS.
Software build	The build number of this software version.
Software release date	The date on which this version of the software was released.
Software name	The internal reference number for this software release.
Software options	The maximum number of calls, and the availability of additional VCS features such as FindMe™, Device Provisioning and Dual Network Interfaces, are controlled through the use of option keys . This section shows all the optional features currently installed on the VCS.
Hardware version	The version number of the hardware on which the VCS software is installed.
Hardware serial number	The serial number of the hardware on which the VCS software is installed.

Time Information

Up time	The amount of time that has elapsed since the system last restarted.
System time (UTC)	The time as determined by the NTP server. If no NTP server has been configured, this will show Time Not Set .
Time zone	The time zone that has been configured on the Time page.
Local time	If an NTP server has been configured, the system time in local time (UTC adjusted according to the local time zone) is shown. If no NTP server has been configured, the time according to the VCS's operating system is shown.

Ethernet status

The **Ethernet status** page provides details of the MAC address and Ethernet speed settings of the VCS.

To view the **Ethernet status** page:

- **Status > System > Ethernet**

The page displays the following information for the LAN 1 port and, if the Dual Network Interfaces option key has been installed, the LAN 2 port:

MAC address	The MAC address of the VCS's Ethernet device for that LAN port.
Speed	The speed of the connection between the LAN port on the VCS and the Ethernet switch.

IP status

The **IP status** page provides details of the IP and DNS settings of the VCS.

To view the **IP status** page:

- **Status > System > IP**

The information shown on this page will vary depending on your system configuration, but may include the following information:

IP

Protocol	Indicates the IP protocol supported by the VCS. IPv4: The VCS will only accept registrations from endpoints using an IPv4 address, and will only take calls between two endpoints or devices communicating via IPv4. It will communicate with other systems via IPv4 only. IPv6: The VCS will only accept registrations from endpoints using an IPv6 address, and will only take calls between two endpoints communicating via IPv6. It will communicate with other systems via IPv6 only. Both: The VCS will accept registrations from endpoints using either an IPv4 or IPv6 address, and will take calls using either protocol. If a call is between an IPv4-only and an IPv6-only endpoint, the VCS will act as an IPv4 to IPv6 gateway (note that this will require a traversal call license). The VCS can communicate with other systems via either protocol.
IPv4 gateway	The IPv4 gateway used by VCS.
IPv6 gateway	The IPv6 gateway used by VCS.
Dual Network Interfaces	Indicates whether the second LAN port has been enabled. This is done by installing the Dual Network Interfaces option key.
LAN 1	Shows the IPv4 address and subnet mask, and IPv6 address of the LAN 1 port.
LAN 2	If the Dual Network Interfaces option key has been installed, this shows the IPv4 address and subnet mask, and IPv6 address of the LAN 2 port.
DNS	
Server 1..5 address	The IP address(es) of each of the DNS servers that will be queried when resolving domain names. Up to 5 DNS servers may be configured.
Domain	Specifies the name to be appended to the host name before a query to the DNS server is executed.

Resource usage

The **Resource usage** page provides statistics about the numbers of current and cumulative calls and registrations on the VCS. This page automatically refreshes every 5 seconds.

To view the **Resource usage** page:

- **Status > System > Resource usage**

The page displays the following information:

Traversal calls	Current: The number of traversal calls going through the VCS at this moment. Max (peak): The highest number of concurrent traversal calls handled by the VCS since it was last restarted. Total: The total number of traversal calls handled by the VCS since it was last restarted.
Non-traversal calls	Current: The number of non-traversal calls going through the VCS at this moment. Max (peak): The highest number of concurrent non-traversal calls handled by the VCS since it was last restarted. Total: The total number of non-traversal calls handled by the VCS since it was last restarted.
Registrations	Current: The number of devices registered to the VCS at this moment. Max (peak): The highest number of devices concurrently registered to the VCS since it was last restarted. Total: The total number of registrations on the VCS since it was last restarted.

Registrations by device

The **Registrations by device** page lists each device currently registered with the VCS, and allows you to remove a device's registration.

To view the **Registrations by device** page:

- **Status > Registrations > By device**

Note that an H.323 device can register with more than one alias; in such cases this page will show only one alias and (when present) one E.164 number for that device.

Note also that a single device can support both the SIP and H.323 protocols; in such a case the SIP registration and the H.323 registration will appear as separate entries on this page.

For a list of all aliases currently registered with the VCS, see the **Registrations by alias** page.

Clicking on a device's **name** or **E164** number takes you to the **Registrations details** page for that device.

The page displays the following information:

Name	For H.323 devices, this is one of its aliases. If the device has registered with more than one alias, this will be (in order of preference) its H.323 ID, URI or email address. For MCUs and Gateways this will be its alias or, if it has not registered an alias, one of its prefixes. For SIP devices, this is its SIP AOR.
E164	For H.323 devices that have registered one or more E.164 numbers, the first will be shown here. For SIP devices this will always be blank because they cannot register E.164 numbers.
Type	Indicates the nature of the registration. This will most commonly be Endpoint, MCU, Gateway, or SIP UA.
Protocol	Whether the registration is for a SIP or H.323 device.
Creation Time	The date and time at which the registration was accepted. If an NTP server has not been configured, this will say Time not set .
IP address	For H.323 devices, this is the RAS address. For SIP UAs it is the Contact address presented in the REGISTER request.

Unregister

Click to remove the selected registrations.

Note that removing a registration will not prevent the same device from automatically re-registering.

Filter

To limit the list of registrations, enter one or more characters in the **Filter** field and select **Filter**. Only those registrations that contain (in any of the displayed fields) the string you entered will be shown. To return to the full list of registrations, click **Reset**.

Registrations by alias

The **Registrations by alias** page lists all the aliases, E.164 numbers and prefixes used by all endpoints and systems currently registered with the VCS.

To view the **Registrations by alias** page:

- **Status > Registrations > By alias**

Note that a single H.323 device can register with more than one alias, and each will appear as a separate entry on this page.

For a list of all devices registered with the VCS, see the **Registrations by device** page.

Clicking on any **Alias** will take you to the **Registrations details** page for the device that registered that alias.

The page displays the following information:

Alias	The H.323 alias, E.164 number, prefix or SIP AOR registered by a device.
Alias type	Shows whether the alias is an H.323 ID, E.164 number, prefix or SIP AOR.
Device type	Indicates the nature of the device that registered the alias. This will most commonly be Endpoint, MCU, Gateway or SIP UA.
Protocol	Indicates whether the registration is for a SIP or H.323 device.
Creation Time	The date and time at which the registration was accepted. If an NTP server has not been configured, this will say Time not set .
IP address	For H.323 devices, this is the RAS address. For SIP UAs it is the Contact address presented in the REGISTER request.

Filter

To limit the list of registrations, enter one or more characters in the **Filter** field and click **Filter**. Only those registrations that contain (in any of the displayed fields) the string you entered will be shown. To return to the full list of registrations, click **Reset**.

Registration history

The **Registration history** page lists all the registrations that are no longer current. It lists the most recent historical registrations since the last restart, up to a maximum of 255.

To view the **Registration history** page:

- **Status > Registrations > History**

The page displays the following information:

Name	The H.323 alias or SIP AOR that the device registered. Clicking on an individual name will take you to the Registrations Details page for that registration.
E164	For H.323 devices that registered one or more E.164 numbers, the first will be shown here. For SIP devices this will always be blank because they cannot register E.164 numbers.
Type	Indicates the nature of the registration. This will most commonly be Endpoint, Gateway, or SIP UA.
Protocol	Indicates whether the registration was for a SIP or H.323 device.
Creation Time	The date and time at which the registration was accepted. If an NTP server has not been configured, this will say Time not set .
End Time	The date and time at which the registration was terminated.
Duration	The length of time that the registration was in place.
Reason	The reason why the registration was terminated.

Filter

To limit the list of registrations, enter one or more characters in the **Filter** field and click **Filter**. Only those registrations that contain (in any of the displayed fields) the string you entered will be shown. To return to the full list of registrations, click **Reset**.

Registration details

The **Registration details** page shows full information about an individual device's registration. The exact details and options shown here will depend on the device's protocol, whether the registration is still current, and whether a Deny List is in use.

To view the **Registration details** page:

- **Status > Registrations > By device**, then click on the device's **Name** or **E164** number
- **Status > Registrations > By alias**, then click on the device's **Alias**
- **Status > Registrations > History**, then click on the device's **Name** or **E164** number.

Unregister

Click to unregister the device. Note that the device may automatically re-register after a period of time, depending on its configuration. To prevent this, you must also [use an Allow List or Deny List](#).

Block

Click to add all the aliases for this registration to the Deny List. (This option is only available if a [Deny List has been activated](#).)

Unregister and block

Click to de-register the device add all the aliases for this registration to the Deny List, thus preventing the device from automatically re-registering. (This option is only available if a [Deny List has been activated](#).)

The links available in the **Related tasks** section may include:

Configure the registration Deny List

Click to go to the **Registration Deny List** page, where you can add entries to the Deny List.

View active calls involving this registration

Click to go to the **Calls** page, filtered for this particular registration.

View previous calls involving this registration

Click to go to the **Call history** page, filtered for this particular registration.

Calls

The **Call status** page lists all the calls currently taking place to or from devices registered with the VCS, or that are passing through the VCS.

To view the **Calls status** page:

- **Status > Calls > Calls**

The page displays the following information:

Start time	The date and time when the call was placed.
Source	The alias of the device that placed the call.
Destination	The alias to which the call was placed. This may be different from the alias that was actually dialed from the device, as it may have been transformed either locally or before the zone was queried.
Bandwidth allocated	The bandwidth allocated to this call. This may be different from the amount requested by the endpoint that placed the call.
Route	The subzone or zone from which the call was received and the subzone or zone to which the call was placed. To see the complete route taken by the call within the VCS, including intermediary subzones, click View to go to the Call summary page.
Protocol	Shows whether the call used H.323, SIP, or both protocols.
Actions	Click View to go to the Call summary page, which lists further details of this call.

Disconnect

Click to disconnect the selected calls.



Call disconnection works differently for H.323 and SIP calls due to differences in the way the protocols work.

For H.323 calls, and interworked H.323 to SIP calls, the **Disconnect** command will actually disconnect the call.

For SIP to SIP calls, the **Disconnect** command will cause the VCS to release all resources used for the call and the call will appear on the system as disconnected. However, SIP calls are peer-to-peer and as a SIP proxy the VCS has no authority over the endpoints. Although releasing the resources may have the side-effect of disconnecting the SIP call, it is also possible that the call signaling, media or both may stay up (depending on the type of call being made). The call will not actually disconnect until the SIP endpoints involved have also cleared their resources.

Filter

To limit the list of calls, enter one or more characters in the **Filter** field and click **Filter**. Only those calls that contain (in any of the displayed fields) the characters you entered will be shown.

To return to the full list of calls, click **Reset**.

Call history

The **Call history** page lists all the calls that are no longer active that have taken place since the VCS was last restarted.

To view the **Call history** page:

- **Status > Calls > History**

The page displays the following information:

Start time	The date and time when the call was placed.
Source	The alias of the device that placed the call.
Destination	The alias to which the call was placed. This may be different from the alias that was actually dialed from the device, as it may have been transformed either locally or before the zone was queried.
Protocol	Shows whether the call used H.323, SIP, or both protocols.
Duration	The length of time of the call.
Status	The reason the call was terminated.
Actions	Click View to go to the Call summary page, which lists further details of this call.

Filter

To limit the list of calls, enter one or more characters in the **Filter** field and click **Filter**. Only calls containing those characters (in any of the displayed fields) are shown.

To return to the full list of calls, click **Reset**.

Call summary

The **Call summary** page (**Status > Calls > Calls** or **Status > Calls > History**, then click **View** for a particular call) provides overview statistics about a call, including information about the most relevant legs.

From here further detailed information about the call can be viewed by using the links in the **Related tasks** section at the bottom of the page:

- **View media statistics for this call** takes you to the **Call media** page, where you can view information about the most relevant session for a call. For traversal calls (where the VCS took the media), it also lists the individual media channels (audio, video, data and so on) that made up the call.
- **View all details of this call** takes you to the **Call details** page, where you can view full information about this call.
- **View search details for this call** takes you to the **Search details** page, which lists full information about all the searches associated with this call's Call Tag, including the subzones and zones that were searched and any transforms that were applied to the alias being searched for.
- **View all events associated with this call** takes you to the **Event Log** page, filtered to show only those events associated with this call's Call Tag.

Searches

About searches

Before a call can be placed, the endpoint being called must be located. The VCS sends and receives a series of messages during its attempt to locate the endpoint being called; these messages are each known as searches. An individual call can have one or more searches associated with it, and these searches can be of different types.

The type of search message that is sent will depend on whether the call is for SIP or H.323, and whether the call request was received locally or from an external zone, as follows:

- For H.323 calls that are placed locally, two messages are sent: the first is an **ARQ** which locates the device being called, and the second is the call **Setup** which sends a request to the device asking it to accept the call. Each message shows up as a separate search in the **Search history** page, but only the **Setup** message will be associated with a particular call.
- For H.323 searches originating from external zones, an **LRQ** will appear in the **Search history** page.
- For SIP, a single message is sent in order to place a call: this will either be a SIP **INVITE** or a SIP **OPTIONS**.



Each search has an individual **Search ID**; each call has an individual **Call Tag** (see [Call IDs, Serial Numbers and Tags](#)).

Search history

The **Search history** page lists the most recent 255 searches that have taken place since the VCS was last restarted.

To view the **Search history** page:

- **Status > Search history**

The page displays the following information:

Start time	The date and time at which the search was initiated.
Search type	The type of message being sent.
Source	The alias of the endpoint that initiated the call.
Destination	The alias that was dialed from the endpoint. This may be different from the alias to which the call was actually placed, as the original alias may have been transformed either locally or before the neighbor was queried.
Found	Indicates whether or not the search was successful.
Actions	Click View to go to the Search details page which lists full details of this call.

Filter

To limit the list of calls, enter one or more characters in the **Filter** field and click **Filter**. Only those calls that contain (in any of the displayed fields) the characters you entered will be shown.

To return to the full list of calls, click **Reset**.

Search details

The **Search details** page (**Status > Search history**, then click **View** for a particular search) lists full information about either an individual search, or all searches associated with a single call (depending on how you reached the page). The information shown includes:

- the subzones and zones that were searched
- the call path and hops
- any transforms that were applied to the alias being searched for

Other information associated with the search and (if it was successful) the resulting call can be viewed using the links in the **Related tasks** section at the bottom of the page.

- **View all events associated with this call tag** takes you to the [Event Log](#) page, filtered to show only those events associated with the Call Tag relating to this search.
- **View call information associated with this call tag** takes you to the [Call summary](#) page, where you can view overview information about the call.
- **View all searches associated with this call tag** will be shown if you are viewing details of an individual search and there are other searches associated with the same call. It takes you to a new **Search details** page which lists full information about all the searches associated with the call's Call Tag.

Local Zone

The **Local Zone status** page lists all the subzones that together make up the Local Zone. This will always include the Default Subzone and the Traversal Subzone, plus any other subzones that have been configured.

To view the **Local Zone** page:

- **Status > Local Zone**



Clicking on a subzone name will take you to the configuration page for that subzone

The page displays the following information:

Subzone name	The names of each subzone currently configured on this VCS.
Registrations	The number of devices currently registered within the subzone. Note that devices cannot be registered to the Traversal Subzone.
Calls	The number of calls currently passing through the subzone. Note that a single call may pass through more than one subzone, depending on the route it takes. For example, traversal calls from a locally registered endpoint will always pass through the Traversal Subzone, so they will show up twice; once in the originating subzone and once in the Traversal Subzone.
Bandwidth used	The total amount of bandwidth used by all calls passing through the subzone.

Zones

The **Zone status** page lists all the zones that are currently configured on your VCS, the number of calls and amount of bandwidth being used by each, and their current status.

The list of zones always includes the Default Zone, plus any other zones that you have created.

To view the **Zones** page:

- **Status > Zones**



Clicking on a zone name takes you to the configuration page for that zone.

Note that this does not apply to the Default Zone, as this is not configurable.

The page displays the following information:

Name	The names of each zone currently configured on this VCS.
Type	The type of zone. See the About zones section for a full description of each zone type.
Calls	The number of calls currently passing out to or received in from each zone.
Bandwidth used	The total amount of bandwidth used by all calls passing out to or received in from each zone.
Status	The current status of each zone.

Links

The **Link status** page lists all the links currently configured on your VCS, along with the number of calls and the bandwidth being used by each link.

To view the **Link status** page:

- **Status > Bandwidth > Links**



Clicking on a link name will take you to the configuration page for that link.

The page displays the following information:

Name	The name of each link.
Calls	The total number of calls currently traversing the link. Note that a single call may traverse more than one link, depending on how your system is configured.
Bandwidth used	The total bandwidth of all the calls currently traversing the link.

Pipes

The **Pipe status** page lists all the pipes currently configured on your VCS, along with the number of calls and the bandwidth being used by each pipe.

To view the **Pipe status** page:

- **Status > Bandwidth > Pipes**



Clicking on a pipe name will take you to the configuration page for that pipe.

The page displays the following information:

Name	The name of each pipe.
Calls	The number of calls currently traversing the pipe. Note that a single call may traverse more than one pipe, depending on how your system is configured.
Bandwidth used	The total bandwidth of all the calls currently traversing the pipe.

TURN relays

The **TURN relays** page lists all the currently active TURN relays on the VCS. For each relay, it shows the requesting client address and port and the corresponding VCS address and port.



TURN services are available on VCS Expressways only. They are configured from the **TURN** page (**VCS configuration > Expressway > TURN**).

To view the **TURN relays** page:

- **Status > TURN relays**

The page displays the following information:

Relay	The index number of the relay.
Address	The IP address and port on the VCS of the relay resource that has been allocated for this particular request.
Client	The IP address and port on the NAT (or the client if there is no NAT) that requested the relay
Creation time	The date and time the relay became active.
Expiry time	The date and time the relay will become inactive.

Click **View** to go to the **TURN relay summary** page where you can see more information about a relay. From here further detailed information about the relay can be viewed by using the links in the **Related tasks** section at the bottom of the page:

- **View permissions for this relay** takes you to the **TURN relay permissions** page, where you can view information about the permissions that have been defined on the relay.
- **View channels for this relay** takes you to the **TURN relay channels** page, where you can view information about the channel bindings that have been defined on the relay.
- **View counters for this relay** takes you to the **TURN relay counters** page, where you can view TURN request, response and error counters, as well as media counters, for the relay.

Presence

The **Status > Applications > Presence** menu has three sub-menus:

- **Publishers**
- **Presentities**
- **Subscribers.**

These pages provide information about endpoints and presentities using the [Presence](#) services on the VCS.

Refer to the [Viewing presence status](#) section for a full explanation of the information on these pages.

OCS Relay

The **OCS Relay status** page lists all the FindMe IDs being handled by the OCS Relay application, and shows the current status of each.

The OCS Relay application is required in deployments that use both Microsoft Office Communicator (MOC) clients and FindMe, if they both use the same SIP domain. Its purpose is to:

- enable the VCS to share FindMe presence information with MOC clients
- enable the Microsoft Office Communications Server (OCS) to forward calls to FindMe IDs.

To view the **OCS Relay status** page:

- **Status > Applications > OCS Relay**



The OCS Relay application is configured from the **OCS Relay** page (**Applications > OCS Relay**).

The page displays the following information:

Alias	The FindMe ID being handled by the OCS Relay application.
Presence state	Shows the presence information currently being published for the FindMe ID.
Registration state	Indicates whether the FindMe ID has registered successfully with OCS. Doing so allows OCS to forward calls to the FindMe ID.
Subscription state	Indicates whether the OCS Relay application has subscribed successfully to the FindMe ID's presence information. Doing so allows MOC clients to view the presence information of FindMe users.

Provisioning

The **Provisioning status** page shows the status of the VCS's provisioning server.

The provisioning server provides basic device provisioning for registered Movi users, without the need for TMS.

To view the **Provisioning status** page:

- **Status > Applications > Provisioning**



The provisioning server is only available if the **Starter Pack** option key is installed.

The page displays the following information:

Server status	The state of the provisioning server.
Subscription counters	A summary of the subscription requests received by the server since the VCS was last restarted. It shows counts of: • the total number of subscription requests received • how many requests were sent a successful provisioning response • failed requests because the account requesting provisioning could not be found • failed requests because the account requesting provisioning had no provisioned devices associated with it

Warnings

The **Warnings** page provides a list of all the warnings currently in place on your system (and, where applicable, their proposed resolution).

Warnings occur when an event or configuration change has taken place on the VCS that requires some manual administrator intervention, such as a restart. Warnings may also be raised for hardware and environmental issues such as faulty disks and fans or high temperatures.

When there are unacknowledged warnings in place on the VCS, a warning icon  appears at the top right of all pages.

To view the **Warnings** page, either:

- click the  icon
- go to **Status > Warnings**

The page displays the following information:

Warning	The warning message.
State	Indicates whether the warning is New or Acknowledged .
Action	Shows you how to address the problem. Clicking the hyperlink either takes you to the appropriate configuration page or displays further detailed instructions.



You should deal with each warning by clicking each **Action** hyperlink and making the necessary configuration changes to resolve the problem.

You can acknowledge a warning by selecting it and clicking on the **Acknowledge** button). This removes the warning icon from the web interface, but the warning will still be listed on the **Warnings** page with a **status** of **Acknowledged**. If a new warning occurs, the warning icon will reappear.

You cannot delete warnings from the **Warnings** page. Warnings are removed by the VCS only after the required action or configuration change has been made.

After a restart of the VCS, any **Acknowledged** warnings that are still in place on the VCS will reappear with a status of **New**, and must be re-acknowledged.

A [list of warnings](#) that can be raised by the VCS is included at the back of this guide.

Hardware

The **Hardware** page provides information about the physical status of your VCS unit.

To view the **Hardware** page:

- **Status > Hardware**

Information displayed includes:

- fan speeds
- component temperatures
- component voltages

Any appropriate minimum or maximum levels are shown to help identify any components operating outside of their standard limits.



Do not attempt to service the apparatus yourself as opening or removing covers may expose you to dangerous voltages or other hazards, and will void the warranty. Refer all servicing to qualified service personnel.

Event Log

The **Event Log** is a list of all the events that have occurred on the system since the last upgrade. The Event Log holds 2GB of data; when this size is reached, the oldest entries are overwritten. However only the first 50MB of event log data can be displayed through the web interface.

To view the **Event Log** page:

- **Status > Logs > Event Log**

To view the Event Log using the CLI:

- `eventlog`

Filtering the Event Log

Search for lets you filter the event log. Enter the words you want to search for and click **Filter**. Only those events that contain all the words you entered are shown.

To do more advanced filtering, click **more options**. This gives you additional filtering methods:

Contains the string: only includes events containing the exact phrase entered here.

Contains any of the words: includes any events that contain at least one of the words entered here.

Not containing any of the words: filters out any events containing any of the words entered here.

Note: use spaces to separate each word you want to filter by.

Click **Filter** to reapply any modified filter conditions. To return to the complete event log listing, click **Reset**.

Reconfigure the log settings

Clicking this link takes you to the **Logging** configuration page. From this page, you can set the level of events that are recorded in the Event Log, and also set up a remote server to which the Event Log can be copied. See the sections on [Setting the Event Log level](#) and [About remote logging](#) for more information.

Results

This section shows all the events, with the most recent being shown first.

Most **tvcs** events contain hyperlinks in one or more of the fields (such fields change color when you hover over them). You can click on the hyperlink to show only those events that contain the same text string.

For example, clicking on the text that appears after **Event=** will filter the list to show all the events of that particular type. Likewise, clicking on a particular **Call-Id** will show just those events that contain a reference to that particular call.

Event Log color coding

Certain events in the Event Log are color-coded so that you can identify them more easily. These events are as follows:

Green

- System Start
- Installation of <item> succeeded
- Registration Accepted
- Call Connected
- Request Successful
- Beginning System Restore
- Completed System Restore

Orange

- System Shutdown

Red

- Registration Rejected
- Registration Refresh Rejected
- Call Rejected
- Security Alert
- License Limit Reached
- Decode Error
- TLS Negotiation Error
- External Server Communications Failure
- Application Failed
- Request Failed
- System Backup Error
- System Restore Error
- Authorization Failure

Event Log

Event Log format

The Event Log is displayed in an extension of the UNIX syslog format:

date time process _name: message _details

where:

Field	Description
date	The local date on which the message was logged.
time	The local time at which the message was logged.
process _name	The name of the program generating the log message. This could include: tvcs for all messages originating from VCS processes findme for FindMe account data migration events web for all web login and configuration events tprovisioning for all events associated with the TMS Agent but will differ for messages from other applications running on the VCS.
message _details	The body of the message (see the Message details field section for further information).

Administrator and FindMe user sessions

Administrator session related events are:

- **Admin Session Start**
- **Admin Session Finish**
- **Admin Session Login Failure**

FindMe user session related events are:

- **User Session Start**
- **User Session Finish**
- **User Session Login Failure**

For both administrator and FindMe user related events, the **Detail** field includes:

- the name of the administrator or FindMe user to whom the session relates, and their IP address
- the date and time that the login was attempted, started, or ended

Event Log

Message details field

For all messages logged from the **tvcs** process, the **message _ details** field, which contains the body of the message, consists of a number of human-readable **name=value** pairs, separated by a space.

The first name element within the **message _ details** field is always **Event** and the last name element is always **Level**.

The table below shows all the possible name elements within the **message _ details** field, in the order that they would normally appear, along with a description of each.



In addition to the events described below, a **syslog.info** event containing the string **MARK** is logged after each hour of inactivity to provide confirmation that logging is still active.

Name	Description
Event	The event which caused the log message to be generated. See Events and levels for a list of all events that are logged by the VCS, and the level at which they are logged.
User	The username that was entered when a login attempt was made.
ipaddr	The source IP address of the user who has logged in.
Protocol	Specifies which protocol was used for the communication. Valid values are: - TCP - UDP - TLS
Reason	Textual string containing any reason information associated with the event.
Service	Specifies which protocol was used for the communication. Will be one of: - H.323 - SIP - H.225 - H.245 - LDAP - Q.931 - NeighbourGatekeeper - Clustering - ConferenceFactory
Message Type	Specifies the type of the message.
Response-code	SIP response code or, for H.323 and interworked calls, a SIP equivalent response code.
Src-ip	Source IP address (the IP address of the device attempting to establish communications). This can be an IPv4 address or an IPv6 address.
Dst-ip	Destination IP address (the IP address of the destination for a communication attempt). The destination IP is recorded in the same format as Src-ip .
Src-port	Source port: the IP port of the device attempting to establish communications.
Dst-port	Destination port: the IP port of the destination for a communication attempt.
Src-alias	If present, the first H.323 alias associated with the originator of the message. If present, the first E.164 alias associated with the originator of the message.
Dst-alias	If present, the first H.323 alias associated with the recipient of the message. If present, the first E.164 alias associated with the recipient of the message.
Detail	Descriptive detail of the Event.

Name	Description
Auth	Whether the call attempt has been authenticated successfully.
Method	SIP method (INVITE, BYE, UPDATE, REGISTER, SUBSCRIBE, etc).
Contact	Contact: header from REGISTER.
AOR	Address of record.
Call-id	The Call-ID header field uniquely identifies a particular invitation or all registrations of a particular client.
Call-serial-number	The VCS-local Call Serial Number that is common to all protocol messages for a particular call.
Tag	The Tag is common to all searches and protocol messages across a VCS network for all forks of a call.
Call-routed	Indicates if the VCS took the signaling for the call.
To	- for REGISTER requests: the AOR for the REGISTER request - for INVITEs: the original alias that was dialed - for all other SIP messages: the AOR of the destination.
Request-URI	The SIP or SIPS URI indicating the user or service to which this request is being addressed.
Num-bytes	The number of bytes sent/received in the message.
Protocol-buffer	Shows the data contained in the buffer when a message could not be decoded.
Duration	Request/granted registration expiry duration.
Time	A full UTC timestamp in YYYY/MM/DD-HH:MM:SS format. Using this format permits simple ASCII text sorting/ordering to naturally sort by time. This is included due to the limitations of standard syslog timestamps.
Level	The level of the event as defined in the About Event Log levels section.
UTCTime	Time the event occurred, shown in UTC format.

Event Log

Events and levels

Event	Description	Level
Admin Session Finish	An administrator has logged off the system.	1
Admin Session Login Failure	An unsuccessful attempt has been made to log in as an administrator. This could be because an incorrect username or password (or both) was entered.	1
Admin Session Start	An administrator has logged onto the system.	1
Application Exit	The VCS application has been exited. Further information may be provided in the Detail event parameter.	1
Application Failed	The VCS application is out of service due to an unexpected failure.	1
Application Start	The VCS has started. Further detail may be provided in the Detail event parameter.	1
Application Warning	The VCS application is still running but has experienced a recoverable problem. Further detail may be provided in the Detail event parameter.	1
Authorization Failure	The user has either entered invalid credentials, does not belong to an access group, or belongs to a group that has an access level of "None". Applies when remote authentication is enabled.	1
Beginning System Backup	A system backup has started.	1
Beginning System Restore	A system restore has started.	1
Call Answer Attempted	An attempt to answer a call has been made.	1
Call Attempted	A call has been attempted.	1
Call Bandwidth Changed	The endpoints in a call have renegotiated call bandwidth.	1
Call Connected	A call has been connected.	1
Call Diverted	A call has been diverted.	1
Call Disconnected	A call has been disconnected.	1
Call Inactivity Timer	A call has been disconnected due to inactivity.	1
Call Rejected	A call has been rejected. The Reason event parameter contains a textual representation of the H.225 additional cause code.	1
Call Rerouted	The VCS has Call Routed mode set to Optimal and has removed itself from the call signaling path.	1
Completed System Backup	A system backup has completed.	1
Completed System restore	A system restore has completed.	1
Configlog Cleared	An operator cleared the Configuration Log.	1
Decode Error	A syntax error was encountered when decoding a SIP or H.323 message.	1
Directory Service Database Started	The TMS Agent database has started.	1

Event Log

Event	Description	Level
Directory Service Database Stopped	The TMS Agent database has stopped.	1
Directory Service Failed Restarting	The TMS Agent failed to restart.	1
Directory Service Restarted	The TMS Agent has restarted.	1
Directory Service Restarting	The TMS Agent is restarting.	1
Directory Service Starting	The TMS Agent is starting.	1
Directory Service Shutting Down	The TMS Agent is shutting down.	1
Error Response Sent	The TURN server has sent an error message to a client (using STUN protocol).	3
Eventlog Cleared	An operator cleared the Event Log.	1
External Server Communication Failure	Communication with an external server failed unexpectedly. The Detail event parameter should differentiate between "no response" and "request rejected". Servers concerned are: • DNS • LDAP servers • Neighbor Gatekeeper • NTP servers • Peers	1
FindMe Search Failed	A search of the FindMe database has failed, for example due to no alias being provided.	1
FindMe Transfer	FindMe user accounts have been migrated across clusters. The Detail event parameter provides additional details.	1
Hardware Failure	There is an issue with the VCS hardware. If the problem persists, contact your Cisco support representative.	1
License Limit Reached	Licensing limits for a given feature have been reached. The Detail event parameter specifies the facility/limits concerned. Possible values for the detail field are: • Non Traversal Call Limit Reached • Traversal Call Limit Reached If this occurs frequently, you may want to contact your Cisco representative to purchase more licenses.	1
Message Received	An incoming RAS message has been received.	2
Message Received	An incoming RAS NSM Keepalive, H.225, H.245 or a RAS message between peers has been received.	3
Message Received	(SIP) An incoming message has been received.	4

Event Log

Event	Description	Level
Message Rejected	This could be for one of two reasons: - The VCS Authentication mode is set to On, and an endpoint has unsuccessfully attempted to send a message (such as a registration request) to the VCS. This could be either because the endpoint has not supplied any authentication credentials, or because its credentials do not match those expected by the VCS. - Clustering is enabled but bandwidth across the cluster has not been configured identically, and the VCS has received a message relating to an unknown peer, link, pipe, subzone or zone.	1
Message Sent	An outgoing RAS message has been sent.	2
Message Sent	An outgoing RAS NSM Keepalive, H.225, H.245 or a RAS message between peers has been sent.	3
Message Sent	(SIP) An outgoing message has been sent.	4
Operator Call Disconnect	An administrator has disconnected a call.	1
Outbound TLS Negotiation Error	The VCS is unable to communicate with another system over TLS. Refer to the event parameters for more information.	1
Policy Change	A policy file has been updated.	1
POST request failed	A HTTP POST request was submitted from an unauthorized session.	1
Provisioning	Diagnostic messages from the provisioning server. The Detail event parameter provides additional information.	1
Reboot Requested	A system reboot has been requested. Refer to the Reason event parameter for specific information.	1
Registration Accepted	A registration request has been accepted.	1
Registration Refresh Accepted	A request to refresh or keep a registration alive has been accepted.	3
Registration Refresh Rejected	A request to refresh a registration has been rejected.	1
Registration Refresh Requested	A request to refresh or keep a registration alive has been received.	3
Registration Rejected	A registration request has been rejected. The Reason and Detail event parameters provide more information about the nature of the rejection.	1
Registration Removed	A registration has been removed by the VCS. The Reason event parameter specifies the reason why the registration was removed. This is one of: - Authentication change - Conflicting zones - Operator forced removal - Operator forced removal (all registrations removed) - Registration superseded.	1
Registration Requested	A registration has been requested.	1
Relay Allocated	A TURN server relay has been allocated.	2
Relay Deleted	A TURN server relay has been deleted.	2

Event Log

Event	Description	Level
Relay Expired	A TURN server relay has expired.	2
Request Failed	A request sent to the Conference Factory has failed.	1
Request Received	A call-related SIP request has been received.	2
Request Received	A non-call-related SIP request has been received.	3
Request Sent	A call-related SIP request has been sent.	2
Request Sent	A non-call-related SIP request has been sent.	3
Request Successful	A successful request was sent to the Conference Factory.	1
Response Received	A call-related SIP response has been received.	2
Response Received	A non-call-related SIP response has been received.	3
Response Sent	A call-related SIP response has been sent.	2
Response Sent	A non-call-related SIP response has been sent.	3
Restart Requested	A system restart has been requested. Refer to the Reason event parameter for specific information.	1
Search Attempted	A search has been attempted.	1
Search Cancelled	A search has been cancelled.	1
Search Completed	A search has been completed.	1
Search Loop detected	The VCS is in Call loop detection mode and has identified and terminated a looped branch of a search.	2
Secure mode disabled	The VCS has successfully exited Advanced account security mode.	1
Secure mode enabled	The VCS has successfully entered Advanced account security mode.	1
Security Alert	A potential security-related attack on the VCS has been detected.	1
Source Aliases Rewritten	A source alias has been changed to indicate the caller's FindMe ID.	1
Success Response Sent	The TURN server has sent a success message to a client (using STUN protocol).	3
System backup completed	The system backup process has completed.	1
System Backup error	An error occurred while attempting a system backup.	1
System backup started	The system backup process has started.	1
System Configuration Changed	An item of configuration on the system has changed. The Detail event parameter contains the name of the changed configuration item and its new value.	1
System restore completed	The system restore process has completed.	1

Event Log

Event	Description	Level
System Restore error	An error occurred while attempting a system restore.	1
System restore started	The system restore process has started.	1
System Shutdown	The operating system was shutdown.	1
System snapshot started	A system snapshot has been initiated.	1
System snapshot completed	A system snapshot has completed.	1
System Start	The operating system has started. The Detail event parameter may contain additional information if there are startup problems.	1
TLS Negotiation Error	Transport Layer Security (TLS) connection failed to negotiate.	1
TMS Agent backup completed	The TMS Agent backup process has completed.	1
TMS Agent backup error	An error occurred while attempting a TMS Agent backup.	1
TMS Agent backup started	The TMS Agent backup process has started.	1
TMS Agent restore completed	The TMS Agent restore process has completed.	1
TMS Agent Restore error	An error occurred while attempting a TMS Agent restore.	1
TMS Agent restore started	The TMS Agent restore process has started.	1
Unregistration Accepted	An unregistration request has been accepted.	1
Unregistration Rejected	An unregistration request has been rejected.	1
Unregistration Requested	An unregistration request has been received.	1
Upgrade	Messages related to the software upgrade process. Refer to the Detail event parameter for specific information.	1
User session finish	A FindMe user has logged out of the system.	1
User session Login failure	An unsuccessful attempt has been made to log in as a FindMe user. This could be because either an incorrect username or password (or both) was entered.	1
User session start	A FindMe user has logged on to the system.	1
Warning acknowledged	An administrator has acknowledged a warning. The Detail event parameter provides information about the nature of the issue.	1
Warning lowered	The issue that caused a warning to be raised has been resolved. The Detail event parameter provides information about the nature of the issue.	1
Warning raised	The VCS has detected an issue and raised a warning. The Detail event parameter provides information about the nature of the issue.	1

Configuration Log

The **Configuration Log** is a list of all changes to the VCS configuration made using the web or command line interface.

The Configuration Log visible using the web interface holds a maximum of 4MB of data; when this size is reached, the oldest entries are overwritten.

To view the **Configuration Log** page:

- **Status > Logs > Configuration Log**

To view the Configuration Log using the CLI:

- **configlog**

Filtering the Configuration Log

Search for lets you filter the Configuration Log. Enter the words you want to search for and click **Filter**. Only those events that contain all the words you entered are shown.

To do more advanced filtering, click **more options**. This gives you additional filtering methods:

Contains the string: only includes events containing the exact phrase entered here.

Contains any of the words: includes any events that contain at least one of the words entered here.

Not containing any of the words: filters out any events containing any of the words entered here.

Note: use spaces to separate each word you want to filter by.

Click **Filter** to reapply any modified filter conditions. To return to the complete Configuration Log listing, click **Reset**.

Results

This section shows all the web-based events, with the most recent being shown first.

Most events contain hyperlinks in one or more of the fields (such fields will change color when you hover over them). You can click on the hyperlink to automatically filter the search so that only those events that contain that same text string are shown.

For example, clicking on the text that appears after **Event=** will filter the list to show all the events of that particular type. Likewise, clicking on a particular **user** will show just those events relating to that particular administrator account.



All events that appear in the Configuration Log are recorded as Level 1 Events, so any changes to the [Logging levels](#) will not affect their presence in the Configuration Log.

Configuration Log events

Changes to the VCS configuration made by administrators using the web interface have an **Event** field of **System Configuration Changed**.

The **Detail** field of each of these events shows:

- the configuration item that was affected
- what it was changed from and to
- the name of the administrator user who made the change, and their IP address
- the date and time that the change was made

System configuration

This section describes all the options that appear under the **System Configuration** menu of the web interface. These options enable you to configure the Cisco VCS in relation to the network in which it is located, for example its IP settings and the external services used by the Cisco VCS (e.g. DNS, NTP and SNMP).



System administration

The **System administration** page allows you to configure the name of the VCS and the means by which it is accessed by administrators.

To go to the **System administration** page:

- **System configuration > System**

To configure these settings using the CLI:

- [xConfiguration SystemUnit Name](#)
- [xConfiguration Administration](#)



You must save your changes and restart the system for any changes (apart from **Session time out**) to take effect.

About the system name

The system name is used to identify the VCS. It appears in various places in the web interface, and in the display on the front panel of the unit (so that you can identify it when it is in a rack with other systems). The system name is also used by Cisco TMS.



We recommend that you give the VCS a name that allows you to easily and uniquely identify it.



If the system name (or IPv6 address) is longer than 16 characters, only the last 16 characters will be shown in the display on the front panel.

About administration access settings

While it is possible to administer the VCS by using a PC connected directly to the unit via a serial cable, you may want to access the system remotely over IP.

You can do this using either or both:

- the [web interface](#) via HTTPS
- a [command line interface](#) via SSH or Telnet.

By default, access via HTTPS and SSH is enabled; access via Telnet is disabled.

You can also enable access via HTTP. However, this mode works by redirecting HTTP calls to the HTTPS port, so HTTPS must also be enabled for access via HTTP to function.

The **Session time out** setting controls whether the system automatically logs out the current user after a configurable period of inactivity.



By default, access via HTTPS and SSH is enabled; access via Telnet is disabled. To securely manage the VCS you should disable Telnet, using the encrypted HTTPS and SSH protocols instead. For further security, disable HTTPS and SSH as well and use the serial port to manage the system.



Cisco TMS accesses the VCS via the web server. If HTTPS mode is turned off, Cisco TMS will not be able to access it.

Ethernet

The **Ethernet** page allows you to configure the speed of the connection between the VCS and the Ethernet switch to which it is connected.

To go to the **Ethernet** page:

- **System configuration > Ethernet**

To configure ethernet settings using the CLI:

- [xConfiguration Ethernet](#)



You must save your changes and restart the system for changes made using this page to take effect.

About Ethernet speed

The **Ethernet speed** setting determines the speed of the connection between the VCS and the Ethernet switch. It must be set to the same value on both systems.

If you have the Dual Network Interfaces option key installed, you will be able to configure the speed for both the LAN 1 and LAN 2 ports separately.

The default is **Auto**, which means that the two systems will auto-negotiate the appropriate speed.



You should only change the default value from **Auto** if the switch to which you are connecting is unable to auto-negotiate. A mismatch in Ethernet speed settings between the VCS and Ethernet switch will at best result in packet loss; at worst it will make the system inaccessible for endpoints and system administrators.

IP

The **IP** page lets you configure the IP protocols and settings of the VCS.

To go to the **IP** page

- **System configuration > IP**

To configure these settings using the CLI:

- [xConfiguration IP](#)
- [xConfiguration IPProtocol](#)
- [xConfiguration Ethernet](#)



You must save your changes and restart the system for changes to take effect.

About IP protocols

You can configure the VCS to use **IPv4**, **IPv6** or **Both** protocols. The default is **Both**.

IPv4: the VCS only: accepts registrations from endpoints using an IPv4 address; takes calls between endpoints communicating via IPv4; communicates with other systems via IPv4.

IPv6: the VCS only: accepts registrations from endpoints using an IPv6 address; takes calls between endpoints communicating via IPv6; communicates with other systems via IPv6.

Both: the VCS accepts registrations from endpoints using either an IPv4 or IPv6 address, and takes calls using either protocol. If a call is between an IPv4-only and an IPv6-only endpoint, the VCS acts as an IPv4 to IPv6 gateway. It communicates with other systems via either protocol.

IPv4 to IPv6 gatewaying (interworking)

The VCS can act as a gateway for calls between IPv4 and IPv6 devices. To enable this feature, select an **IP protocol** of **Both**.

Calls for which the VCS is acting as an IPv4 to IPv6 gateway are traversal calls and require a traversal call license.



Some endpoints support both IPv4 and IPv6, but an endpoint can use only one protocol when registering with the VCS.

Whether the endpoint uses IPv4 or IPv6 is determined by the IP addressing scheme used on the endpoint to specify the IP address of the VCS. After the endpoint has registered using either IPv4 or IPv6, the VCS only sends calls to it using this addressing scheme. Calls made to that endpoint from another device using the other addressing scheme are converted (gatewayed) by the VCS.

External LAN interface

The **External LAN interface** field indicates which LAN port is connected to your external network.

It also determines the port from which TURN server relay allocations are made.

About IP routes (static routes)

You can set the default IPv4 and IPv6 gateways used by the VCS. These are the gateways to which IP requests are sent for IP addresses that do not fall within the VCS's local subnet. However, you can also configure additional IP routing information (static routes) on the VCS. This is sometimes required when using the Dual Network Interfaces option and occasionally required in other complex network deployments. You can configure routes for up to 50 networks and host combinations.

IP routes can be configured using the CLI only:

- [xConfiguration IP Route](#)
- [xCommand RouteAdd](#)

About LAN configuration

LAN 1 is the primary network port on the VCS. You can configure the **IPv4 address** and **subnet mask**, and **IPv6 address** for this port. For VCS Expressway boxes behind a static NAT, you can also configure the NAT's IP address. If you have Dual Network Interfaces installed, you can also configure these options for the LAN 2 port.

The VCS is shipped with a default IP address of 192.168.0.100 (for both LAN ports). This lets you connect the VCS to your network and access it via the default address so that you can configure it remotely.

About Dual Network Interfaces

The Dual Network Interface option key enables the LAN 2 port on the VCS Expressway for both management and call signaling. This allows you to have a second IP address for your VCS.

This configuration is intended for high-security deployments where the VCS is located in a DMZ between two separate firewalls on separate network segments. In such deployments, routers prevent devices on the internal network from being able to route IP traffic to the public internet, and instead the traffic must pass through an application proxy such as the VCS.

To enable this feature you must purchase and install the appropriate option key. Contact your Cisco representative for information.



You should configure the LAN 1 port and restart the VCS before configuring the LAN 2 port.



If you have Dual Network Interfaces enabled but only want to configure one of the Ethernet ports, you must use LAN 1.

About static NAT

It is possible to deploy a VCS Expressway behind a static NAT device, allowing it to have separate public and private IP addresses. This feature is intended for use in deployments where the VCS Expressway is located in a DMZ, and has the Dual Network Interfaces feature enabled (see the previous section).

In these deployments, the externally-facing LAN port has static NAT enabled in order to use both a private and public IPv4 address; the internally facing LAN port does not have static NAT enabled and uses a single IPv4 (or IPv6) address.

In such a deployment, when configuring traversal clients to use the VCS Expressway as a traversal server, it is the latter internally-facing IP address of the VCS Expressway that should be used.

To enable the use of a static NAT:

- ensure that the Dual Network Interfaces option key is installed
- on the **IP** page (**System Configuration > IP**), for the externally-facing LAN port:
 - in the **IPv4 address** field, enter the VCS Expressway's private IP address
 - select an **IPv4 static NAT mode** of **On**
 - in the **IPv4 static NAT** field, enter the VCS Expressway's public IP address - this is the IP address of the outside of the NAT.



The static NAT configuration options are only available on VCS Expressway systems which have the Dual Network Interfaces option key installed.

Quality of Service (QoS)

The **Quality of Service** (QoS) page lets you configure QoS options for outbound traffic from the VCS.

This allows the network administrator to tag all signaling and media packets flowing through the VCS with one specific QoS tag and hence provide the ability to prioritize video traffic over normal data traffic. Management traffic, for example SNMP messages, is not tagged.

To go to the **Quality of Service** page:

- **System configuration > Quality of Service**

To configure QoS settings using the CLI:

- [xConfiguration IP QoS](#)

Supported mechanisms

The VCS supports the **DiffServ** (Differentiated Services) mechanism which puts the specified **Tag value** in the TOS (Type Of Service) field of the IPv4 header or TC (Traffic Class) field of the IPv6 header.



You must save your changes and restart the system for changes to take effect.

DNS

The **DNS** page lets you configure the VCS's DNS servers and DNS settings.

To go to the **DNS** page:

- **System configuration > DNS**

To configure DNS settings using the CLI:

- [xConfiguration IP DNS](#)

About DNS servers

You must specify at least one DNS server to be queried for address resolution if you want to either:

- use FQDNs (Fully Qualified Domain Names) instead of IP addresses when specifying external addresses (for example for LDAP and NTP servers, neighbor zones and peers)
- use features such as [URI dialing](#) or [ENUM dialing](#)

You can specify up to 5 DNS servers. The VCS sends requests to all configured servers in parallel taking the first result received and discounting the rest.



This can lead to confusing behavior should local network administrators, for example, deploy "split horizon" DNS where records held on an internal, corporate, DNS server use the same domain names but with different values to those on the public internet - an often used tactic in corporate intranets.

About DNS settings

Local host name

The **Local host name** defines the DNS host name that this VCS is known by.

- It must be unique for each peer in a cluster.
- It is used to identify the VCS on a remote log server (a default name of "TANDBERG" is used if the **Local host name** is not specified).

Domain name

The **Domain name** is used when attempting to resolve unqualified server addresses (for example "ldap" or "ldap_server"). It is appended to the unqualified server address before the query is sent to the DNS server. If the server address is fully qualified (for example "ldap.server") or is in the form of an IP address, the domain name is not appended to the server address before querying the DNS server.

It applies only to the following configuration settings in the VCS:

- LDAP server
- NTP server
- External Manager server
- Remote logging server

You are recommended to use an IP address or FQDN (Fully Qualified Domain Name) for all server addresses.

The DNS **Domain name** plays no part in URI dialing.



Note that the FQDN of the VCS is the **Local host name** plus the **Domain name**.

Time

The **Time** page lets you configure the VCS's NTP server and specify your local time zone.

To go to the **Time** page:

- **System configuration > Time**

To configure these settings using the CLI:

- [xConfiguration NTP Address](#)
- [xConfiguration TimeZone Name](#)

About the NTP server

The NTP server is a remote server with which the VCS synchronizes to ensure its time setting is accurate. The NTP server provides the VCS with UTC time.

Accurate timestamps play an important part in H.323 [authentication](#), helping to guard against replay attacks. For this reason, if you are using authentication in a deployment that includes H.323, both the VCS and the endpoints must use an NTP server to synchronize their system time.

Accurate time is also vital for configuration replication to work properly if the VCS is in a cluster, and to ensure accurate timestamps in system logs.



Traversal clients must always authenticate with traversal servers, even if the server's **Authentication Mode** is **Off**. Therefore, for a traversal client and traversal server to connect to each other, both must be configured with details of an NTP server.



SIP-only deployments do not require the use of NTP for authentication.

To configure the VCS with an NTP server, on the **Time** page enter the IP address or FQDN (or server address, if a DNS **Domain Name** has also been configured) of the **NTP server** to be used when synchronizing system time.

- The **NTP server** field defaults to one of four NTP servers provided by Cisco, either: *0.ntp.tandberg.com*, *1.ntp.tandberg.com*, *2.ntp.tandberg.com* or *3.ntp.tandberg.com*.
- The connection status to the NTP server is shown in the **Status** area.

VCS time display

Local time is used throughout the web interface. It is shown in the system information bar at the bottom of the screen and is used to set the timestamp that appears at the start of each line in the Event Log and Configuration Log.

Internally, the VCS maintains its system time in UTC. It is based on the VCS's operating system time, which is synchronized using an NTP server if one is configured. If no NTP server is configured, the VCS uses its own operating system time to determine the time and date.

Specifying your local **Time zone** lets the VCS determine the local time where the system is located. It does this by offsetting UTC time by the number of hours associated with the selected time zone. It also adjusts the local time to account for summer time (also known as daylight saving time).

The **Time** page displays both UTC and local time in the **Status** area.

Note that a UTC system timestamp is included at the end of each entry in the Event Log and Configuration Log.

Login page

The **Login page** allows you to specify a message and image that will appear to users attempting to log in to the VCS.

To go to the **Login page** page:

- **System configuration > Login page**

This feature is not configurable using the CLI.

The welcome message **title** and **text** will appear to administrators when attempting to log in using the CLI, and to FindMe users and administrators when attempting to log in using the web interface.

The **Image** will appear to FindMe users and administrators when attempting to log in using the web interface.

SNMP

The **SNMP** page lets you configure the VCS's SNMP settings:

To go to the SNMP page:

- **System configuration > SNMP**

To configure SNMP settings using the CLI:

- [xConfiguration SNMP](#)

About SNMP

Tools such as Cisco TMS or HP OpenView may act as SNMP Network Management Systems (NMS). They allow you to monitor your network devices, including the VCS, for conditions that might require administrative attention.

The VCS supports the most basic MIB-II tree (.1.3.6.1.2.1) as defined in [RFC 1213 \[23\]](#). The information made available by the VCS includes the following:

- system uptime
- system name
- location
- contact
- interfaces
- disk space, memory, and other machine-specific statistics

To allow the VCS to be monitored by an SNMP NMS (including Cisco TMS), you must **Enable** SNMP on the VCS and provide the name of the **SNMP community** within which it resides. You may optionally provide the name of a **System contact** and the physical **Location** of the system for reference by administrators when following up on queries.

By default, SNMP is **Disabled** with a **SNMP community name** of **public**.



SNMP is disabled by default because of the potentially sensitive nature of the information involved. Do not enable SNMP on a VCS on the public internet or in any other environment where you do not want to expose internal system information.



The VCS does not support SNMP traps or SNMP sets, therefore it cannot be managed using SNMP.

External manager

About external managers

An external manager is a remote system, such as the Cisco TelePresence Management Suite (Cisco TMS), used to monitor events occurring on the VCS, for example call attempts, connections and disconnections. The use of an external manager is optional.

Configuration

The **External manager** page allows you to configure the VCS's external manager settings.

To go to the **External manager** page:

- **System configuration > External manager**

To configure external manager settings using the CLI:

- [xConfiguration ExternalManager](#)

The options are:

Address and path

To use an external manager, you must configure the VCS with the IP address or host name and path of the external manager to be used.

If you are using Cisco TMS as your external manager, use the default path of **tms/public/external/management/SystemManagementService.asmx**.

Protocol

This setting allows you to determine whether communications with the external manager are over **HTTP** or **HTTPS**.

Certificate verification mode

You may also require that the VCS verifies the certificate presented by the external manager by selecting a **Certificate verification mode** of **On**.

If you do this, you must also add the certificate of the issuer of the external manager's certificate to the file containing the VCS's trusted CA certificates. This is done from the **Security certificates** page (**Maintenance > Security certificates**).

Logging

Overview

The VCS provides an Event Logging facility for troubleshooting and auditing purposes. The Event Log records information about such things as calls, registrations, and messages sent and received.

The **Logging** page lets you:

- specify the **Log level** to set the amount of information to record
- copy the Event Log to a **Remote syslog server**

To go to the **Logging** page:

- **System configuration > Logging**

To view the Event Log using the web interface:

- **Status > Logs > Event Log**

To view the Event Log using the CLI:

- `eventlog`

About Event Log levels

All events have an associated level in the range 1-4, with Level 1 Events considered the most important. The table below gives an overview of the levels assigned to different events.



See the [Events and levels](#) section for a complete list of all events that are logged by the VCS, and the level at which they are logged.

Level	Assigned Events
Level 1	High-level events such as registration requests and call attempts. Easily human readable. For example: • call attempt/connected/disconnected • registration attempt/accepted/rejected
Level 2	All Level 1 Events, plus: • logs of protocol messages sent and received (SIP, H.323, LDAP etc.) excluding noisy messages such as H.460.18 keepalives and H.245 video fast-updates
Level 3	All Level 1 and Level 2 Events, plus: • protocol keepalives • call-related SIP signaling messages
Level 4	The most verbose level: all Level 1, Level 2 and Level 3 Events, plus: • network level SIP messages

Setting the Event Log level

You can control which events are logged by the VCS by setting the log level. All events with a level numerically equal to and lower than the specified logging level are recorded in the Event Log. So, at Level 1, only Level 1 events are logged; at Level 2, both Level 1 and Level 2 events are logged, and so on. The default is **1**. To set the log level:

- **System configuration > Logging**
- [xConfiguration Log Level](#)



Logging at level 3 or level 4 is not usually recommended as the Event Log holds a maximum of 2GB of data and logging at these levels on a busy system could cause the Event Log to be recycled too quickly.



Changes to the Event Log level affect both the Event Log that you can view using the web interface, and the information that is copied to the [remote log server](#) (if any) that you have configured.



Changes to the Event Log level are not retrospective. If you change the Event Log level, it will only affect what is logged from that point onwards.

About remote logging

The Event Log is always stored locally on the VCS. However, it is often convenient to collect copies of all Event Logs from various systems in a single location. A computer running a BSD-style syslog server, as defined in [RFC 3164 \[4\]](#), may be used as the central log server. Note that:

- A VCS will not act as a central logging server for other systems.
- Events are always logged locally (to the Event Log) regardless of whether or not remote logging is enabled.
- The VCS may use any of the 23 available syslog facilities for different messages. Specifically, LOCAL0..LOCAL7 (facilities 16..23) are used by different components of the application software on the VCS.

Enabling remote logging

To enable remote logging, you must configure the VCS with the address of the central log server to which the Event Log will be copied. To do this:

- **System configuration > Logging**
- [xConfiguration Log Server Address](#)

Cisco VCS configuration

This section provides information on the pages that appear under the **Protocols**, **Registrations** and **Authentication** sub-menus of the **VCS Configuration** menu. These pages allow you to configure the functionality of the Cisco VCS in each of these areas.

This section includes the following information:

- [an overview of H.323 and the H.323 configuration options available on the VCS](#)
- [an overview of SIP and the SIP configuration options available on the VCS](#)
- [how to configure the VCS to act as a SIP to H.323 gateway](#)
- how to control registrations on the VCS using [authentication](#) and [Allow Lists and Deny Lists](#).



H.323 overview

About H.323 on the VCS

The VCS supports the H.323 protocol: it is an H.323 gatekeeper. It will also provide [interworking](#) between H.323 and SIP, translating between the two protocols to enable endpoints that only support one of these protocols to call each other. In order to support H.323, the **H.323 mode** must be enabled.

Using the Cisco VCS as an H.323 gatekeeper

As an H.323 gatekeeper, the VCS accepts registrations from H.323 endpoints and provides call control functions such as address translation and admission control.

To enable the VCS as an H.323 Gatekeeper, you must ensure that **H.323 mode** is set to **On** (**VCS configuration > Protocols > H.323**).



This is the default setting, so the VCS will work as an H.323 gatekeeper "out of the box", without any other special configuration.

H.323 endpoint registration

H.323 endpoints in your network must register with the VCS in order to use it as their gatekeeper.

There are two ways an H.323 endpoint can locate a VCS with which to register: manually or automatically. The option is configured on the endpoint itself under the **Gatekeeper Discovery** setting (consult your endpoint manual for how to access this setting).

- If the mode is set to automatic, the endpoint will try to register with any VCS it can find. It does this by sending out a Gatekeeper Discovery Request, to which eligible VCSs will respond.
- If the mode is set to manual, you must specify the IP address of the VCS with which you wish your endpoint to register, and the endpoint will attempt to register with that VCS only.

Configuring H.323

The **H.323** page allows you to enable and disable H.323 system-wide on the VCS, and configure H.323-specific ports and settings.

To go to the **H.323** page:

- **VCS configuration > Protocols > H.323**

To configure H.323 using the CLI:

- [xConfiguration H323](#)

Enabling H.323

H.323 support is **On** by default.

To enable or disable H.323 on the VCS, use the **H.323 Mode** setting.

Configuring H.323 ports

The default VCS configuration uses standard port numbers so you can use H.323 services out of the box without having to first set these up.

You can configure the following H.323 ports:

Registration UDP port

The listening port for H.323 UDP registrations. Default is 1719.

Call signaling TCP port

The listening port for H.323 call signaling. Default is 1720.

Call signaling port range start

Specifies the lower port in the range used by H.323 calls after they are established. Default is 15000.

Call signaling port range end

Specifies the upper port in the range used by H.323 calls after they are established. Default is 19999.



The call signalling port range must be great enough to support all the required concurrent calls.

Registration conflict mode

An H.323 endpoint may attempt to register with the VCS using an alias that has already been registered on the VCS from another IP address. The reasons for this could include:

- Two endpoints at different IP addresses are attempting to register using the same alias.
- A single endpoint has previously registered using a particular alias. The IP address allocated to the endpoint then changes, and the endpoint attempts to re-register using the same alias.

You can determine how the VCS will behave in this situation by configuring the **Registration conflict mode**. The options are:

Reject: denies the new registration.

Overwrite: deletes the original registration and replaces it with the new registration.

The default is **Reject**.

Time to live

H.323 endpoints must periodically re-register with the VCS in order to confirm that they are still functioning. The VCS allows you to configure the interval (in seconds) between these re-registrations, known as the **Time to live**. The default is 1800.



Some older endpoints do not support the ability to periodically re-register with the system. In this case, and in any other situation where the system has not had a confirmation from the endpoint within the specified period, it will send an IRQ to the endpoint to verify that it is still functioning.

Call time to live

When the endpoint is in a call, the VCS will periodically poll it to confirm whether it is still in the call. If the endpoint does not respond, the call will be disconnected.

The VCS allows you to configure the interval (in seconds) at which the endpoints are polled, known as the **Call time to live**. The default is 120.



The system will poll endpoints in a call regardless of whether the call type is traversal or non-traversal.

Auto discover

The VCS has an **Auto discover** setting which determines whether it will respond to [Gatekeeper Discovery Requests](#) sent out by endpoints. The default is **On**.

To prevent H.323 endpoints being able to register automatically with the VCS, set **Auto discover** to **Off**. This will mean that endpoints will be able to register with the VCS only if they have been configured with the VCS's IP address.

Caller ID

You can specify whether the **Caller ID** displayed on the destination endpoint includes or excludes the prefix of the ISDN gateway when displaying the caller's E.164 number.



Including the prefix allows the recipient to directly return the call.

SIP overview

About SIP on the Cisco VCS

The VCS supports the SIP protocol. It can act as a:

- SIP registrar
- SIP proxy
- SIP Presence Server

The VCS can provide interworking between SIP and H.323, translating between the two protocols to enable endpoints that only support one of these protocols to call each other.

To support SIP, [SIP mode must be enabled](#) and at least one of the SIP transport protocols (UDP, TCP or TLS) must be active.

Using the Cisco VCS as a SIP registrar

For a SIP endpoint to be contactable via its registered alias, it must register its location with a SIP registrar. The VCS can act as a SIP registrar for up to 20 domains.

SIP aliases always take the form **username@domain**. To make the VCS act as a SIP registrar, you must [configure it with the SIP domains](#) for which it will be authoritative. It will then accept registration requests for any endpoints attempting to register with an alias that includes that domain.

If no domains are configured, the VCS will not act as a SIP registrar but it may proxy the request to another registrar depending upon the **SIP registration proxy mode** setting.

Using the Cisco VCS as a SIP Presence Server

The VCS supports the SIP-based SIMPLE protocol. It can act as a:

- Presence Server
- Presence User Agent

for any of the [SIP domains](#) for which it is authoritative.

For full information on how to use the VCS as a SIP Presence server, see the [Presence](#) section.

Using the Cisco VCS as a SIP proxy server

The VCS can act as a SIP proxy server when **SIP mode** is enabled. The role of a proxy server is to forward requests (such as REGISTER and INVITE) from endpoints or other proxy servers on to further proxy servers or to the destination endpoint.

The VCS's behavior as a SIP proxy server is determined by:

- the **SIP registration proxy mode** setting
- the presence of Route Set information in the request header
- whether the proxy server from which the request was received is a neighbor of the VCS

A Route Set specifies the path to take when requests are proxied between an endpoint and its registrar. For example, when a REGISTER request is proxied by a VCS, the VCS adds a path header component to the request. This signals that calls to that endpoint should be routed through the VCS. This is usually required in situations where firewalls exist and the signalling must follow a specified path to successfully traverse the firewall. For more information about path headers, see [RFC 3327 \[10\]](#).

When the VCS proxies a request that contains Route Set information, it forwards it directly to the URI specified in the path. Any call processing rules configured on the VCS are bypassed. This may present a security risk if the information in the Route Set cannot be trusted. For this reason, you can configure how the VCS proxies requests that contain Route Sets by setting the **SIP registration proxy mode** as follows:

- **Off**: requests containing Route Sets are rejected. This setting provides the highest level of security.
- **Proxy to known only**: requests containing Route Sets are proxied only if the request was received from a known zone.
- **Proxy to any**: requests containing Route Sets are always proxied.



In all cases, requests that do not have Route Sets are proxied as normal in accordance with existing call processing rules.

This setting only applies to dialog-forming requests, such as INVITE and SUBSCRIBE. Other requests, such as NOTIFY, are always proxied regardless of this setting.

Proxying registration requests

If the VCS has no SIP domains configured, or it receives a registration request for a domain for which it is not acting as a Registrar, then the VCS may proxy the registration request. This depends on the **SIP registration proxy mode** setting, as follows:

- **Off**: the VCS does not proxy any registration requests. They are rejected with a "403 Forbidden" message.
- **Proxy to known only**: the VCS proxies the request in accordance with existing call processing rules, but only to known neighbor, traversal client and traversal server zones.
- **Proxy to any**: this is the same as **Proxy to known only** but for all zone types i.e. it also includes ENUM and DNS zones.

Accepting proxied registration requests

If the VCS receives a proxied registration request, in addition to the VCS's standard [registration controls](#), you can also control whether the VCS accepts the registration depending upon the zone through which the request was received. You do this through the **Accept proxied registrations** setting when [configuring a zone](#).



Proxied registrations are classified as belonging to the zone they were last proxied from. This is different from non-proxied registration requests which are assigned to a subzone within the VCS.

SIP endpoint registration

A SIP endpoint registers with the VCS (or another SIP registrar) to identify its current location and thus enable it to receive calls.

To register your SIP endpoint with a SIP registrar, enter the IP address or FQDN of the SIP registrar into your SIP endpoint.

To use the VCS as the SIP registrar for a particular endpoint, the VCS must be configured with the SIP domain used by the endpoint (**VCS configuration > Protocols > SIP > Domains**).

Movi v2.0 (or later) clients

As for any other SIP endpoint, the VCS acts as a SIP registrar and SIP proxy for Movi™ v2.0 (or later) clients - no other special support or configuration is required on the VCS.

Configuring SIP

The **SIP** page is used to enable and disable SIP system-wide on the VCS, and to configure SIP-specific ports and settings.

To go to the **SIP** page:

- **VCS configuration > Protocols > SIP > Configuration**

To configure SIP using the CLI:

- [xConfiguration SIP](#)

Enabling SIP

SIP support is **On** by default.

To enable and disable SIP functionality (i.e. SIP registrar and SIP proxy services) on the VCS, use the **SIP mode** setting.

SIP registration expiry

SIP endpoints must periodically re-register with the SIP registrar in order to prevent their registration expiring. The **Registration expire delta** setting defines the interval within which SIP endpoints must register with the VCS. The default is **60**.



This applies only to endpoints registered with the VCS. It does not apply to endpoints whose registrations are proxied through the VCS.

SIP registration proxy mode

This specifies how proxied registrations and requests containing Route Sets are handled.

Off: registration requests are not proxied (but are still permitted locally if the VCS is authoritative for that domain). Requests with existing Route Sets are rejected.

Proxy to known only: registration requests are proxied in accordance with existing call processing rules, but only to known neighbor, traversal client and traversal server zones. Requests containing Route Sets are proxied only if they were received from a known zone.

Proxy to any: registration requests are proxied in accordance with existing call processing rules to all known zones. Requests containing Route Sets are always proxied.

The default is **Off**.

SIP protocols and ports

The VCS supports SIP over UDP, TCP and TLS transport protocols. You can configure whether or not incoming and outgoing connections using each protocol are supported, and if so, the ports on which the VCS listens for such connections. This is done using the **Mode** and **Port** settings for each protocol.

The default **Mode** for each protocol is **On**; the default ports are:

- UDP port: 5060
- TCP port: 5060
- TLS port: 5061



At least one of the transport protocols must be set to a **Mode** of **On** for SIP functionality to be supported.

You can also specify the range of ports the VCS uses when TCP and TLS connections are established by configuring the **TCP outbound port start** and **TCP outbound port end** settings.

The default range is **25000** to **29999**. This range must be sufficient to support all required concurrent connections.

Session timers

You can configure SIP session expiry timers through the **Session refresh interval** and **Minimum session refresh interval** settings.

For more information on session timers, refer to [RFC 4028 \[14\]](#).

SIP device interoperability

You can configure the following device interoperability settings:

Require UDP BFCP mode

Controls whether the VCS requires the use of the **com.tandberg.udp.bfcp** extension for endpoints that support it.

Require Duo Video mode

Controls whether the VCS requires the use of the **com.tandberg.sdp.duo.enable** extension for endpoints that support it.



The default settings for these modes are not supported by some neighbor systems so make sure you select the appropriate [Zone profile](#) when configuring zones.

SIP domains

The **Domains** page lists the SIP domains for which the VCS is authoritative, and allows you to add, edit and delete SIP domains.

To go to the **Domains** page:

- **VCS configuration > Protocols > SIP > Domains.**

To configure SIP domains using the CLI:

- [xCommand DomainAdd](#)
- [xCommand DomainDelete](#)
- [xConfiguration SIP Domains](#)

The VCS will act as a SIP registrar and Presence Server for all these domains, and will accept registration requests for any SIP endpoints attempting to register with an alias that includes any of these domains.

Overview

The VCS is able to act as a gateway between SIP and H.323, translating calls from one protocol to the other. This is known as “interworking”.

By default, the VCS acts as a SIP-H.323 and H.323-SIP gateway but only if at least one of the endpoints that are involved in the call is locally registered. You can change this setting so that the VCS will act as a SIP-H.323 gateway regardless of whether the endpoints involved are locally registered. You also have the option to disable interworking completely.



The VCS uses the protocol of the incoming call when searching a zone for a given alias. If the search is unsuccessful the VCS may then search the same zone again using the alternative protocol, depending on where the search came from and the **Interworking mode** (**VCS configuration > Protocols > Interworking**).

If the request has come from a neighboring system and **Interworking mode** is set to **RegisteredOnly**, the VCS will search the Local Zone using both protocols, and all other zones using the native protocol only (because it will interwork the call only if one of the endpoints is locally registered). If **Interworking mode** is set to **On**, or the request has come from a locally registered endpoint, the VCS will search the Local Zone and all external zones using both protocols.



Calls for which the VCS acts as a SIP to H.323 gateway are traversal calls. They will therefore require a traversal call license.

Configuring interworking

The **Interworking** page is used to configure whether or not the VCS acts as a gateway between SIP and H.323 calls.

To go to the **Interworking** page:

- **VCS configuration > Protocols > Interworking**

To configure interworking using the CLI:

- [xConfiguration Interworking Mode](#)

H.323 <-> SIP interworking mode

The options for this setting are:

Off: the VCS will not act as a SIP-H.323 gateway.

RegisteredOnly: the VCS will act as a SIP-H.323 gateway but only if at least one of the endpoints is locally registered.

On: the VCS will act as a SIP-H.323 gateway regardless of whether the endpoints are locally registered.



You are recommended to leave this setting as **RegisteredOnly** (where calls are interworked only if at least one of the endpoints is locally registered). Unless your network is correctly configured, setting it to **On** (where all calls can be interworked) may result in unnecessary interworking, for example where a call between two H.323 endpoints is made over SIP, or vice versa.

Enabling SIP endpoints to dial H.323 numbers

SIP endpoints can only make calls in the form of URIs - e.g. **name@domain**. If the caller does not specify a domain when placing the call, the SIP endpoint will automatically append its own domain to the number that is dialed. So if you dial **123** from a SIP endpoint, the search will be placed for **123@domain**. If the H.323 endpoint being dialed is just registered as **123**, then the VCS won't be able to locate the alias **123@domain** and the call will fail.

The solutions are to either:

1. Ensure all your endpoints, both H.323 and SIP, register with an alias in the form **name@domain**.
2. Create a pre-search transform on the VCS that strips the **@domain** portion of the alias for those URIs that are in the form of **number@domain**.

Refer to the [Pre-search transforms](#) section for information on how to configure pre-search transforms, and to the [Stripping @domain for dialing to H.323 numbers](#) section for an example of how to do this.

Registration overview

Endpoint registration

For an endpoint to use the VCS as its H.323 gatekeeper or SIP registrar, the endpoint must first register with the VCS. The VCS can be configured to control which devices are allowed to register with it. Two separate mechanisms are provided:

- a [device authentication process](#) based on the username and password supplied by the endpoint
- a simple Registration Restriction Policy that uses [Allow Lists or Deny Lists](#) to specify which aliases can and cannot register with the VCS, and the ability to control registrations based on IP addresses and subnet ranges through the specification of subzone membership rules and [subzone registration policies](#).

It is possible to use both mechanisms together. For example, you can use authentication to verify an endpoint's identity from a corporate directory, and registration restriction to control which of those authenticated endpoints may register with a particular VCS.

This section gives an overview of how endpoints and other devices register with the VCS, and then describes the two mechanisms by which registrations can be restricted.



For specific information about how registrations are managed across peers in a cluster, see the [Sharing registrations across peers](#) section.

Registrations on a Cisco VCS Expressway

If a traversal-enabled endpoint registers directly with a VCS Expressway, the VCS Expressway will provide the same services to that endpoint as a VCS Control, with the addition of firewall traversal. Traversal-enabled endpoints include all Cisco TelePresence Expressway™ endpoints and third party endpoints which support the ITU H.460.18 and H.460.19 standards.

Endpoints that are not traversal-enabled can still register with a VCS Expressway, but they may not be able to make or receive calls through the firewall successfully. This will depend on a number of factors:

- whether the endpoint is using SIP or H.323
- the endpoint's position in relation to the firewall
- whether there is a NAT in use
- whether the endpoint is using a public IP address

For example, if an endpoint is behind a NAT or firewall, it may not be able to receive incoming calls and may not be able to receive media for calls it has initiated. SIP endpoints can also work behind a NAT but can only receive video if they send it as well.

To ensure firewall traversal will work successfully for H.323 endpoints behind a NAT, the endpoint must be traversal-enabled.

MCU, gateway and Content Server registration

H.323 systems such as gateways, MCUs and Content Servers can also register with a VCS. They are known as locally registered services. These systems are configured with their own prefix, which they provide to the VCS when registering. The VCS will then know to route all calls that begin with that prefix to the gateway, MCU or Content Server as appropriate. These prefixes can also be used to control registrations.

SIP devices cannot register prefixes. If your dial plan dictates that a SIP device should be reached via a particular prefix, then you should add the device as a neighbor zone with an associated search rule using a pattern match equal to the prefix to be used.



The Cisco TelePresence MPS 200 and MPS 800, and the Cisco TelePresence Content Server both support Expressway. They can therefore register directly with a VCS Expressway for firewall traversal.

Registration overview

Finding a Cisco VCS with which to register

Before an endpoint can register with a VCS, it must determine which VCS it can or should be registering with. This setting is configured on the endpoint, and the process is different for SIP and H.323.

SIP

SIP endpoints must find a SIP Registrar with which to register. The SIP registrar maintains a record of the endpoint's details against the endpoint's Address of Record (AOR). When a call is received for that AOR, the SIP registrar refers to the record in order to find the endpoint to which it corresponds. (Note that the same AOR can be used by more than one SIP endpoint at the same time.)

The SIP registrar will only accept registrations for domains for which it is authoritative.

There are two ways a SIP endpoint can locate a registrar with which to register: manually or automatically. The option is configured on the endpoint itself under the **SIP Server Discovery** option (consult your endpoint user guide for how to access this setting; it may also be referred to as **Proxy Discovery**).

- If the **Server Discovery** mode is set to automatic, the endpoint will send a REGISTER message to its SIP Server. This will be forwarded (via DNS if necessary) to the registrar that is authoritative for the domain with which the endpoint is attempting to register. For example, if an endpoint is attempting to register with a URI of **john.smith@example.com**, the request will be sent to the registrar authoritative for the domain **example.com**.
- If the **Server Discovery** mode is set to manual, the user must specify the IP address or FQDN of the registrar with which they wish to register, and the endpoint will attempt to register with that registrar only.

The VCS is a SIP Server for endpoints in its local zone, and can also act as a SIP registrar.

- If the VCS is acting as the endpoint's SIP Server and SIP registrar, when the registration request is received from the endpoint it will be accepted by the VCS and the endpoint will be registered and able to receive inbound calls. See the [Using the Cisco VCS as a SIP registrar](#) section for more information.
- If the VCS is acting as the endpoint's SIP server but is not a SIP registrar, it will proxy the registration request. See the [Proxying registration requests](#) section for more information.

H.323

There are two ways an H.323 endpoint can locate a VCS with which to register: manually or automatically. The option is configured on the endpoint itself under the **Gatekeeper Discovery** setting (consult your endpoint manual for how to access this setting).

- If the mode is set to automatic, the endpoint will try to register with any VCS it can find. It does this by sending out a Gatekeeper Discovery Request, to which eligible VCSs will respond.
- If the mode is set to manual, you must specify the IP address of the VCS with which you wish your endpoint to register, and the endpoint will attempt to register with that VCS only.

Preventing automatic registrations

You can prevent H.323 endpoints being able to register automatically with the VCS by disabling **Auto Discovery** on the VCS. This setting is accessed using:

- **VCS configuration > Protocols > H.323.**
You will be taken to the **H.323** page.
- [xConfiguration H323 Gatekeeper AutoDiscovery](#)

The **Auto Discovery** setting determines whether the VCS responds to the Gatekeeper Discovery requests sent out by endpoints.

The options are as follows:

On: the VCS will respond to Gatekeeper discovery requests.

Off: the VCS will reject Gatekeeper discovery requests. H.323 endpoints will be able to register with the VCS only if their **Gatekeeper Discovery** setting is **Manual** and they have entered the IP address of the VCS.

Device authentication

The **Device authentication configuration** page controls whether systems attempting to communicate with the VCS must authenticate with it first, and if so, the type of database used by the VCS to store the authentication credentials used by these systems.

To go to the **Device authentication configuration** page:

- **VCS configuration > Authentication > Devices > Configuration**

To configure authentication using the CLI:

- [xConfiguration Authentication](#)

Authentication mode

The VCS can be configured to use a username and password-based challenge-response scheme to determine whether it will permit communications from other systems. This process is known as authentication, and is controlled using the **Authentication mode** setting.

The options are:

On: systems attempting to communicate with the VCS, including endpoints attempting to send registration requests to the VCS, must first authenticate with it.

For H.323, any credentials in the message are checked against the authentication database. The message is allowed if the credentials match, or if there are no credentials in the message. For SIP, any messages originating from an endpoint in a local domain will be authenticated.

Off: incoming messages are not authenticated.

The default is **Off**.



Accurate timestamps play an important part in authentication, helping to guard against replay attacks. For this reason, if you are using authentication, both the VCS and the endpoints must use an NTP server to synchronize their system time. See the [About the NTP server](#) section for information on how to configure this for the VCS.

Authentication database

When **Authentication mode** is **On**, endpoints must authenticate with the VCS before they can register. In order to authenticate successfully, the endpoint must supply the VCS with a username. For Cisco endpoints using H.323, the username is the endpoint's **Authentication ID**; for Cisco endpoints using SIP it is the endpoint's **Authentication username**.



For details of how to configure endpoints with a username and password, please consult the endpoint manual.

To verify the identity of the device, the VCS needs access to a database on which all authentication credential information (usernames, passwords, and other relevant information) is stored. This database may be located either locally on the VCS, or on an LDAP Directory Server. The VCS looks up the endpoint's username in the database and retrieves the authentication credentials for that entry. If the credentials match those supplied by the endpoint, the registration is allowed to proceed.

The **Database type** setting determines which database the VCS will use during authentication:

Local database: the local authentication database is used. You must [configure the local authentication database](#) to use this option.

LDAP database: a remote LDAP database is used. You must [configure the LDAP server](#) to use this option.

The default is **LocalDatabase**.



If the VCS is a traversal server, you must ensure that each traversal client's authentication credentials are entered into the selected database.



The VCS supports the [ITU H.235 specification \[1\]](#) for authenticating the identity of H.323 network devices with which it communicates.

Device authentication using LDAP

Overview

If the VCS is using an LDAP server for authentication, the process is as follows:

1. The endpoint presents its username and authentication credentials (these are generated using its password) to the VCS, and the aliases with which it wants to register.
2. The VCS looks up the username in the LDAP database and obtains the authentication and alias information for that entry.
3. If the authentication credentials match those supplied by the endpoint, the registration will continue.

The VCS then determines which aliases the endpoint is allowed to attempt to register with, based on the **alias origin** setting. For H.323 endpoints, you can use this setting to override the aliases presented by the endpoint with those in the H.350 directory, or you can use them in addition to the endpoint's aliases. For SIP endpoints, you can use this setting to reject a registration if the endpoint's AOR does not match that in the LDAP database.

Configuring the LDAP server directory

The directory on the LDAP server should be configured to implement the [ITU H.350 specification \[2\]](#) to store credentials for devices with which the VCS communicates. The directory should also be configured with the aliases of endpoints that will register with the VCS.



See the [LDAP configuration for device authentication](#) appendix for instructions on configuring LDAP servers.

Configuring LDAP server settings

The **Device LDAP Configuration** page is used to configure a connection to the LDAP database for device authentication.

To go to the **Device LDAP Configuration** page:

- **VCS configuration > Authentication > Devices > LDAP configuration**

To configure these settings using the CLI:

- [xConfiguration LDAP](#)
- [xConfiguration Authentication LDAP](#)

LDAP server

The IP address or FQDN (or server address, if a DNS **Domain Name** has also been configured) of the LDAP server.

Port

The IP port of the LDAP server.

The default is **389**.

Encryption

Determines whether the connection to the LDAP server is encrypted using Transport Layer Security (TLS).

TLS: TLS encryption is used for the connection to the LDAP server.

Off: no encryption is used.

The default is **Off**.

The link **Upload a CA Certificate file for TLS** takes you to the **Security certificates** page, where you can upload a file containing the trusted CA certificate for the LDAP server. This is required for encrypted connections between the VCS and the LDAP server. See the [Security certificates](#) section for more information.

User DN

The user distinguished name used by the VCS when binding to the LDAP server.

Password

The password used by the VCS when binding to the LDAP server.

Base DN

The area of the directory on the LDAP server to search for credential information. This should be specified as the Distinguished Name (DN) in the LDAP directory under which the H.350 objects reside.

Alias origin

This setting determines the aliases with which the endpoint will attempt to register. The options are:

LDAP: for SIP registrations the AOR presented by the endpoint is registered providing it is listed in the LDAP database for the endpoint's username.

For H.323 registrations:

- At least one of the aliases presented by the endpoint must be listed in the LDAP database for that endpoint's username. If none of the presented aliases are listed it is not allowed to register.
- The endpoint will register with all of the aliases (up to a maximum of 20) listed in the LDAP database. Aliases presented by the endpoint that are not in the LDAP database will not be registered.
- If no aliases are listed in the LDAP database, the endpoint will register with all the aliases it presented.
- If no aliases are presented by the endpoint, it will register with all the aliases listed in the LDAP database for its username.

MCUs are treated as a special case. They register with the presented aliases and ignore any aliases in the LDAP database. (This is to allow MCUs to additively register aliases for conferences.)

Combined: the aliases presented by the endpoint are used in addition to any listed in the LDAP database for the endpoint's username. In other words, this is the same as for **LDAP**, except that if an endpoint presents an alias that is not in the LDAP database, it will be allowed to register with that alias.

Endpoint: the aliases presented by the endpoint are used; any in the LDAP database are ignored. If no aliases are presented by the endpoint, it is not allowed to register.

The default is **LDAP**.



To use the LDAP database for device authentication, you must also go to the **Device authentication configuration** page and select a **Database type** of **LDAP database**.

Authentication using a local database

Overview

The local authentication database is included as part of your VCS system. The database can hold up to 2,500 entries, each consisting of a name and password.

Name

The username used by the endpoint when authenticating with the VCS.

Password

The password used by the endpoint when authenticating with the VCS.

Configuring the local authentication database

The **Local authentication database** page lists and allows you to manage the credentials stored in the local authentication database.

To go to the **Local authentication database** page:

- **VCS configuration > Authentication > Devices > Local database**

To manage the local authentication database using the CLI:

- [xConfiguration Authentication Credential](#)
- [xCommand CredentialAdd](#)
- [xCommand CredentialDelete](#)



You can sort these entries by clicking on the **Name** column heading.

View/Edit

Click **View/Edit** to make changes to an existing entry. You are taken to the **Edit credential** page.

Delete

Click **Delete** to remove a credential from the list.

New

Select **New** to add a new entry to the local authentication database. You are taken to the **Create credential** page.



The same credentials can be used by more than one endpoint - you do not need to have a separate entry in the database for each endpoint.

Authenticating with external systems

Outbound connection credentials

The **Outbound connection credentials** page is used to configure a username and password that the VCS will use whenever it is required to authenticate with external systems. For example, when the VCS is forwarding an invite from an endpoint to another VCS, that other system may have authentication enabled and will therefore require your local VCS to provide it with a username and password.

Note that these settings are not used by traversal client zones. Traversal clients, which must always authenticate with traversal servers before they can connect, configure their connection credentials per [traversal client zone](#).

To go to the **Outbound connection credentials** page:

- **VCS configuration > Authentication > Outbound connection credentials**

To configure authentication using the CLI:

- [xConfiguration Authentication Password](#)
- [xConfiguration Authentication UserName](#)

Registering aliases

About alias registration

After the [authentication](#) process (if required) has been completed, the endpoint will then attempt to register its alias(es) with the VCS.

H.323

When registering, the H.323 endpoint presents the VCS with one or more of the following:

- one or more H.323 IDs
- one or more E.164 aliases
- one or more URIs

Users of other registered endpoints can then call the endpoint by dialing any of these aliases.



You are recommended to register your H.323 endpoints using a URI. This facilitates interworking between SIP and H.323, as SIP endpoints register using a URI as standard.



You are recommended to not use aliases that reveal sensitive information. Due to the nature of H.323, call setup information is exchanged in an unencrypted form.

SIP

When registering, the SIP endpoint presents the VCS with its contact address (IP address) and logical address (Address of Record). The logical address is considered to be its alias, and will generally be in the form of a URI.

Attempts to register using an existing alias

An endpoint may attempt to register with the VCS using an alias that is already registered to the system. How this is managed depends on how the VCS is configured and whether the endpoint is SIP or H.323.

H.323

An H.323 endpoint may attempt to register with the VCS using an alias that has already been registered on the VCS from another IP address. The reasons for this could include:

- two endpoints at different IP addresses are attempting to register using the same alias
- a single endpoint has previously registered using a particular alias. The IP address allocated to the endpoint, or the port the endpoint uses to communicate with the VCS, then changes, and the endpoint is attempting to re-register using the same alias.

You can determine how the VCS will behave in this situation by configuring the **Registration Conflict Mode**, using either:

- **VCS configuration > Protocols > H.323**

You will be taken to the **H.323** page.

- [xConfiguration H323 Gatekeeper Registration ConflictMode](#)

The options are:

Reject: the registration from the new IP address will be rejected. This is useful if your priority is to prevent two users registering with the same alias.

Overwrite: the existing registration will be overwritten using the new IP address. This is useful if your network is such that endpoints are often allocated new IP addresses, because it will prevent unwanted registration rejections.

The default is **Reject**.

SIP

A SIP endpoint will always be allowed to register using an alias that is already in use from another IP address. When a call is received for this alias, all endpoints registered using that alias will be called simultaneously. This SIP feature is known as “forking”.

Blocking registrations

If you have [configured the VCS to use a Deny List](#), from the **Registration details** page (**Status > Registrations > by Device** and click a **Name**; or **Status > Registrations > by Alias** and click an **Alias**; or **Status > Registrations > History** and click an **Alias**) you will have an option to block the registration. This will add all the aliases used by that endpoint to the Deny List.

Allow and Deny Lists

About Allow and Deny Lists

When an endpoint attempts to register with the VCS it presents a list of aliases. You can control which endpoints are allowed to register by setting the **Restriction Policy** to **Allow List** or **Deny List** and then including any one of the endpoint's aliases on the Allow List or the Deny list as appropriate. Each list can contain up to 2,500 entries.

When an endpoint attempts to register, each of its aliases is compared with the patterns in the relevant list to see if it matches. Only one of the aliases needs to appear in the Allow List or the Deny List for the registration to be allowed or denied.

For example, If the Registration Restriction policy is set to **Deny List** and an endpoint attempts to register using three aliases, one of which matches a pattern on the Deny List, that endpoint's registration will be denied. Likewise, if the Registration Restriction policy is set to **Allow List**, only one of the endpoint's aliases needs to match a pattern on the Allow List for it to be allowed to register using all its aliases.

Allow Lists and Deny Lists are mutually exclusive: only one may be in use at any given time.



You can also control registrations at the [subzone](#) level. Each subzone's registration policy can be configured to allow or deny registrations assigned to it via the subzone membership rules.

Activating use of Allow or Deny Lists

The **Registration Configuration** page allows you to specify whether an Allow List or a Deny List should be used when determining which endpoints may register with the VCS.

To go to the **Registration Configuration** page:

- **VCS configuration > Registration > Configuration.**

To configure this using the CLI:

- [xConfiguration Registration RestrictionPolicy](#)

The **Restriction policy** option specifies the policy to be used when determining which endpoints may register with the VCS. The options are:

None: any endpoint may register.

AllowList: only those endpoints with an alias that matches an entry in the Allow List may register.

DenyList: all endpoints may register, unless they match an entry on the Deny List.

The default is **None**.



If you have elected to use an Allow List or a Deny List, you must also go to the appropriate configuration page (**VCS configuration > Registration > Allow List** or **VCS configuration > Registration > Deny List**) to create the list to be used.

Removing existing registrations

After an Allow List or Deny List has been activated, it controls all registration requests from that point forward. However, any existing registrations may remain in place, even if the new list would otherwise block them. Therefore, you are recommended to manually remove all existing unwanted registrations after you have implemented an Allow List or Deny List.

To manually remove a registration, go to **Status > Registrations > By device**, select the registration(s) you want to remove, and click **Unregister**.

Re-registrations

All endpoints must periodically re-register with the VCS in order to keep their registration active. If you do not manually delete the registration, the registration could be removed when the endpoint attempts to re-register, but this depends on the protocol being used by the endpoint:

- H.323 endpoints may use "light" re-registrations which do not contain all the aliases presented in the initial registration, so the re-registration may not get filtered by the Allow List or Deny List. If this is the case, the registration will not expire at the end of the registration timeout period and must be removed manually.
- SIP re-registrations contain the same information as the initial registrations so will be filtered by the Allow List and Deny List. This means that, after the list has been activated, all SIP registrations will disappear at the end of their registration timeout period.

The frequency of re-registrations is determined by the **Registration Expire Delta** setting for SIP (**VCS configuration > Protocols > SIP > Configuration**) and the **Time to Live** setting for H.323 (**VCS configuration > Protocols > H.323**).

Allow and Deny Lists

Using the Allow and Deny Lists

Allow List

The **Registration Allow List** page lets you view and manage the entries in the Allow List.

To go to the **Registration Allow List** page:

- **VCS configuration > Registration > Allow List.**

To manage the Allow List using the CLI:

- [xCommand AllowListAdd](#)
- [xConfiguration Registration AllowList](#)

Deny List

The **Registration Deny List** page lets you view and manage the entries in the Deny List.

To go to the **Registration Deny List** page:

- **VCS configuration > Registration > Deny List.**

To manage the Deny List using the CLI:

- [xCommand DenyListAdd](#)
- [xConfiguration Registration DenyList](#)

Managing entries in the Allow and Deny Lists

The **Registration Allow List** and **Registration Deny List** pages both work in the same way:

Pattern

The pattern against which an alias is compared.

Type

The way in which the **Pattern** must match the alias for the registration to be allowed/denied. Options are:

Exact: the alias must match the **Pattern** exactly.

Prefix: the alias must begin with the **Pattern**.

Suffix: the alias must end with the **Pattern**.

Regex: the **Pattern** is a regular expression. See the [Regular expression reference](#) Appendix for further information.



You can test whether a pattern matches a particular alias by using the [Check pattern](#) page (**Maintenance > Tools > Check pattern**).

Description

An optional free-form description of the entry.

View/Edit

Click **View/Edit** to make changes to an existing entry.

New

Click **New** to add a new entry to the list.

Delete

Click **Delete** to remove the pattern from the list.



After configuring the Allow or Deny List, you must set the restriction policy (**VCS configuration > Registration > Configuration**) to either **Allow List** or **Deny List** for it to be activated.

Zones and neighbors

This section begins with an [overview](#) of all the different types of subzones and zones and how these fit into the overall structure of your video communication network.

It then provides information on the pages that appear under the **Local Zone** and **Zones** sub-menus of the VCS Configuration menu.

These pages describe how to:

- [configure the VCS's Local Zone](#) (which is made up of subzones, including the Traversal Subzone and Default Subzone)
- [create and configure external zones](#) to communicate with other systems and endpoints, including other VCSs, Gatekeepers, Border Controllers or SIP devices, and endpoints contactable via DNS or ENUM dialing
- design a [dial plan](#)



About your video communications network

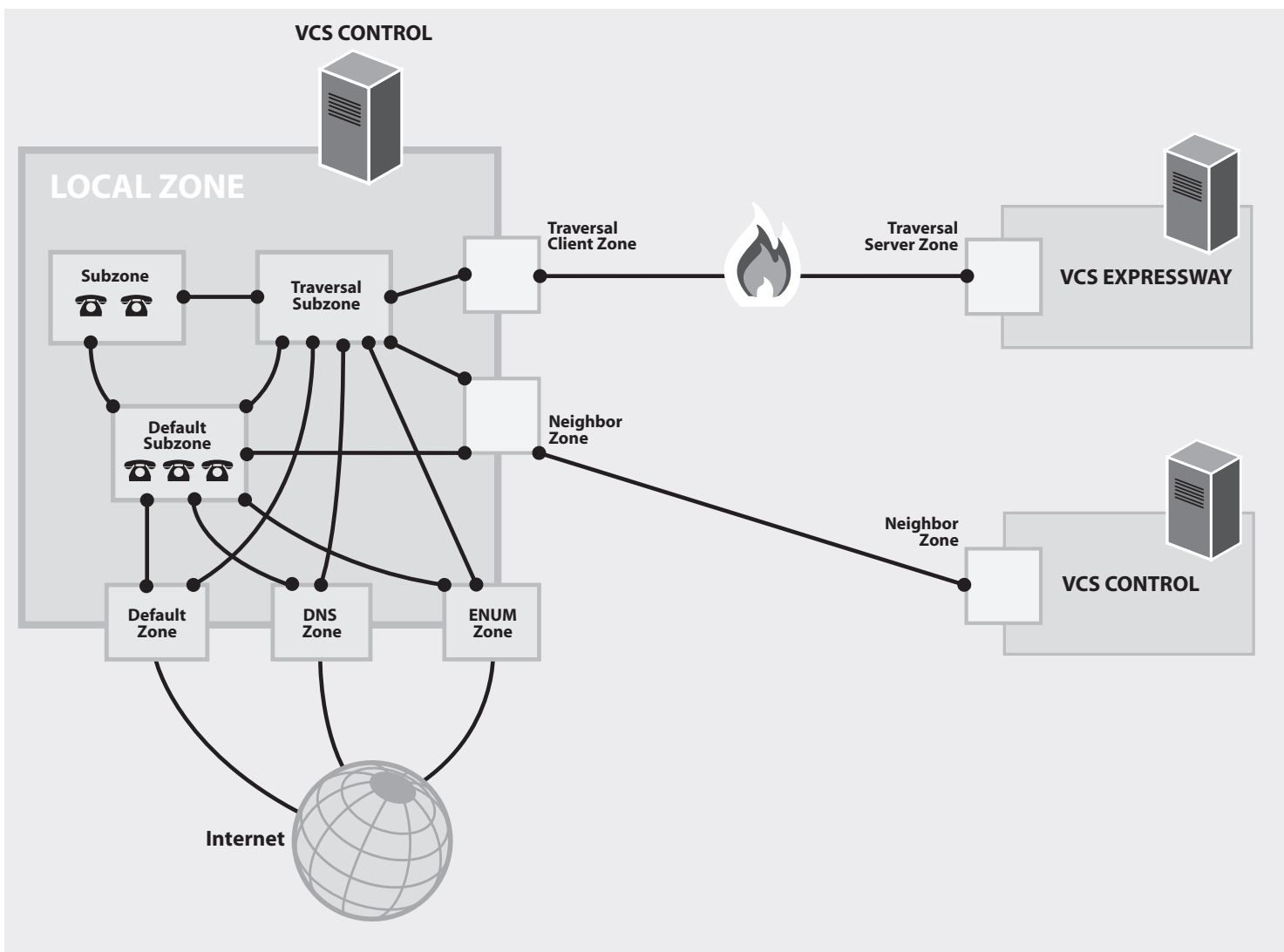
The most basic implementation of a video communications network is a single Cisco VCS connected to the internet with one or more endpoints registered to it. However, depending on the size and complexity of your enterprise the VCS may be part of a network of endpoints, other VCSs and other network infrastructure devices, with one or more firewalls between it and the internet. In such situations you may wish to apply restrictions to the amount of bandwidth used by and between different parts of your network.

This section will give you an overview of the different parts of the video communications network and the ways in which they can be connected. This information should allow you to configure your VCS to best suit your own infrastructure.

Example network diagram

The diagram opposite shows the different components of a VCS (i.e. subzones and zones) and how they interrelate. Using a VCS Control as the example Local Zone, it shows that it is made up of a number of subzones which are all connected by links. The Local Zone is also connected to external VCSs and to the internet via different types of zones.

All these components are described in more detail in the sections that follow.



Overview

The collection of all endpoints, gateways, MCUs and Content Servers registered with the VCS make up its **Local Zone**.

The Local Zone is divided into **subzones**. These include an automatically created **Default Subzone** and up to 1000 manually configurable subzones.

When an endpoint registers with the VCS it is allocated to an appropriate subzone based on **subzone membership rules**. These rules specify the range of IP addresses or alias pattern matches for each subzone. If an endpoint's IP address or alias does not match any of the membership rules, it is assigned to the Default Subzone.

The Local Zone may be independent of network topology, and may comprise multiple network segments.

The VCS also has two special types of subzones:

- the **Traversal Subzone**, which is always present (see the [About the Traversal Subzone](#) section for more information)
- the **Cluster Subzone**, which is always present but only used when your VCS is part of a cluster (see the [Cluster Subzone](#) section for more information)

Configuring the Local Zone and its subzones

Bandwidth management

The Local Zone's subzones are used for bandwidth management. After you have set up your subzones you can apply bandwidth limits to:

- individual calls between two endpoints within the subzone
- individual calls between an endpoint within the subzone and another endpoint outside of the subzone
- the total of calls to or from endpoints within the subzone

For full details of how to create and configure subzones, and apply bandwidth limitations to subzones including the Default Subzone and Traversal Subzone, see the chapter on [Bandwidth control](#).

Local Zone searches

One of the functions of the VCS is to route a call received from a locally registered endpoint or external zone to its appropriate destination. Calls are routed based on the address or alias of the destination endpoint.

When searching for a destination endpoint, the VCS searches its Local Zone and external zones. You can prioritize the order in which these zones are searched, and filter the search requests sent to each zone, based on the address or alias being searched for. This allows you to reduce the potential number of search requests sent to the Local Zone and out to external zones, and speed up the search process.

For further information on how to configure search rules for the Local Zone, see the [Configuring search and zone transform rules](#) section.

Traversal Subzone

The Traversal Subzone is a conceptual subzone; no endpoints can be registered to it, but all traversal calls (calls for which the VCS takes the media in addition to the signaling) pass through it. The Traversal Subzone's purpose is to control the amount of bandwidth used by traversal calls, as these can be particularly resource-intensive. See the chapter on [Bandwidth control](#) and the section on [Bandwidth consumption of traversal calls](#) for more information on controlling the bandwidth of traversal calls.

What are traversal calls?

A traversal call is any call passing through the VCS that includes both the signaling (information about the call) and media (voice and video). The only other type of call is a non-traversal call, where the signaling passes through the VCS but the media goes directly between the endpoints (or between one endpoint and another VCS in the call route, or between two VCSs in the call route).

The following types of calls require the VCS to take the media. They are classified as traversal calls and will always pass through the Traversal Subzone:

- firewall traversal calls, where the local VCS is either the traversal client or traversal server
- calls that are gatewayed (interworked) between H.323 and SIP on the local VCS
- calls that are gatewayed (interworked) between IPv4 and IPv6 on the local VCS
- for VCSs with Dual Network Interfaces enabled, calls that are inbound from one LAN port and outbound on the other
- a SIP to SIP call when one of the participants is behind a NAT (unless both endpoints are using [ICE](#) for NAT traversal)

All such calls require a traversal call license each time they pass through the Traversal Subzone.

Traversal calls use more resource than non-traversal calls, and the numbers of each type of call are licensed separately. The VCS has one license for the maximum number of concurrent traversal calls it can take, and another for the maximum number of concurrent non-traversal calls. You can increase the number of each type of call available on your VCS by purchasing and installing the appropriate [option key](#).

Note that a non-traversal call on a VCS Expressway will consume a traversal license if there are no non-traversal call licenses available.



A call is “traversal” or “non-traversal” from the point of view of the VCS through which it is being routed at the time. A call between two endpoints may pass through two or more VCSs. Some of these VCSs may just take the signaling, in which case the call will be a non-traversal call for that VCS. Other VCSs in the route may need to take the media as well, and so the call will count as a traversal call on that particular VCS.

Configuring the Traversal Subzone ports

The VCS allows you to configure the range of ports used for the media in traversal calls. A single traversal call can consist of up to 5 types of media (audio, video, far end camera control, dual streams and BFCP) and each type of media may require a pair of ports – for example, audio and video each require one port for RTP, and one for RTCP. Separate pairs of ports are required for the inbound and outbound portions of a call. A single traversal call can therefore take up to 20 ports.

The default range for the ports to be used for media is 50000 - 52399 UDP, but these can be changed to any values between 1024 and 65533. Ports are allocated from this range in pairs, with the first port number of each pair being an even number. Therefore the range must start with an even number and end with an odd number.

To configure the ports used for media in traversal calls:

- **VCS configuration > Local Zone > Traversal Subzone**
- [xConfiguration Traversal Media Port Start](#)
- [xConfiguration Traversal Media Port End](#)



You must ensure that the port range is large enough to support the maximum number of traversal calls available on your VCS. A single traversal call can take up to 20 ports (5 pairs in each direction). So for example, if your VCS is licensed for 5 traversal calls you must ensure that the range of ports configured for traversal media is at least 100. If you add extra traversal calls to your system, you must also ensure that the range of ports available is sufficient.

About zones

A zone is a collection of endpoints, either all registered to a single system (for example a Cisco VCS, gatekeeper, or Border Controller), or located in a certain way such as via an ENUM or DNS lookup.

Zones are used to:

- control through links whether calls can be made between your local subzones and these other zones
- manage the bandwidth of calls between your local subzones and endpoints in other zones
- search for aliases that are not registered locally

You can configure up to 1000 zones of five different types. The VCS also has a non-configurable Default Zone.

See the [Zone configuration](#) sections for information on the configuration options available for all zone types.

See the [Configuring search and zone transform rules](#) section for information on including zones as targets for search rules.

Neighbor zone

A neighbor zone could be a collection of endpoints registered to another system (such as a VCS, gatekeeper, or Border Controller), or it could be a SIP device (for example Microsoft Office Communications Server (OCS) 2007). The other system or SIP device is referred to as a neighbor. Neighbors can be part of your own enterprise network, part of a separate network, or even standalone systems.

You create a neighbor relationship with the other system by adding it as a neighbor zone on your local VCS. After you have added it, you can:

- query the neighbor about its endpoints
- apply transforms to any requests before they are sent to the neighbor
- control the bandwidth used for calls between your local VCS and the neighbor zone

See the [Configuring neighbor zones](#) section for information on the specific configuration options available.



Neighbor zone relationship definitions are one-way. Adding a system as a neighbor to your VCS does not automatically make your VCS a neighbor of that system.



Inbound calls from any configured neighbor are identified as coming from that neighbor.

Traversal client zone

To traverse a firewall, the VCS must be connected with a traversal server (for example a VCS Expressway or a TANDBERG Border Controller).

In this situation your local VCS is a traversal client, so you create a connection with the traversal server by creating a traversal client zone on your local VCS. You then configure the client zone with details of the corresponding zone on the traversal server. (The traversal server must also be configured with details of the VCS client zone.)

After you have neighbored with the traversal server you can:

- use the neighbor as a traversal server
- query the traversal server about its endpoints
- apply transforms to any queries before they are sent to the traversal server
- control the bandwidth used for calls between your local VCS and the traversal server

See the [Configuring traversal client zones](#) section for information on the specific configuration options available.



Traversal client-server zone relationships must be two-way. For firewall traversal to work, the traversal server and the traversal client must each be configured with the other's details. (See the [Quick guide to traversal client - server configuration](#) section for more information.) The client and server will then be able to communicate over the firewall and query each other.

Traversal server zone

A VCS Expressway is able to act as a traversal server, providing firewall traversal on behalf of traversal clients (for example, VCS Controls or gatekeepers).

In order to act as a traversal server, the VCS Expressway must have a special type of two-way relationship with each traversal client. To create this connection, you create a traversal server zone on your local VCS Expressway and configure it with the details of the corresponding zone on the traversal client. (The client must also be configured with details of the VCS Expressway.)

After you have neighbored with the traversal client you can:

- provide firewall traversal services to the traversal client
- query the traversal client about its endpoints
- apply transforms to any queries before they are sent to the traversal client
- control the bandwidth used for calls between your local VCS and the traversal client

See the [Configuring traversal server zones](#) section for information on the specific configuration options available.

ENUM zone

ENUM zones allow you to locate endpoints via an ENUM lookup. You can create one or more search rules for ENUM zones based on the ENUM DNS suffix used and/or by pattern matching of the endpoints' aliases.

After you have configured one or more ENUM zones, you can:

- apply [transforms](#) to alias search requests directed to that group of endpoints
- control the bandwidth used for calls between your local VCS and each group of ENUM endpoints

See the [Configuring ENUM zones](#) section for information on the configuration options available.



See the [ENUM dialing](#) section for more information on the use of ENUM zones.

DNS zone

DNS zones allow you to locate endpoints via a DNS lookup. You can create one or more search rules for DNS zones based on pattern matching of the endpoints' aliases.

After you have configured one or more DNS zones, you can:

- apply [transforms](#) to alias search requests directed to that group of endpoints
- control the bandwidth used for calls between your local VCS and each group of DNS endpoints

See the [Configuring DNS zones](#) section for information on the configuration options available.

Default Zone

Any incoming calls from endpoints or other devices that are not recognized as belonging to any of the existing configured zones are deemed to be coming from the Default Zone.

The VCS comes pre-configured with the Default Zone and default links between it and both the Default Subzone and the Traversal Subzone.

The purpose of the Default Zone is to manage incoming calls from unrecognized endpoints to the VCS. You can do this by:

- deleting the default links; this prevents any incoming calls from unrecognized endpoints
- applying pipes to the default links; this lets you control the bandwidth consumed by incoming calls from unrecognized endpoints

The Default Zone has no configuration options.



The default links can be reinstated at any time by using the command:

[`xCommand DefaultLinksAdd`](#)

Zone configuration

Overview

To neighbor with another system (such as another VCS or gatekeeper), create a connection over a firewall to a traversal server or traversal client, or discover endpoints via an ENUM or DNS lookup, you must configure a zone on the local VCS.

When adding a new zone you must specify its **Type**. The zone type indicates the nature of the connection and determines which configuration options are available. For traversal server zones, traversal client zones and neighbor zones this includes providing information about the neighbor system such as its IP address and ports.

The **Zones** page lists all the zones that have been configured on the VCS, and lets you add, edit or delete zones.

To go to the **Zones** page:

- **VCS configuration > Zones.**

Click on the zone you want to configure (or click **New** to create a new zone, or click **Delete** to remove a zone).

To add a new zone using the CLI:

- [xCommand ZoneAdd](#)

To configure existing zones using the CLI:

- [xConfiguration Zones Zone \[1..1000\]](#)

The following sections describe the various zone configuration settings that can be applied.

TLS certificate verification of neighbor systems

When a SIP TLS connection is established between a VCS and a neighbor system, the VCS can be configured to check the X.509 certificate of the neighbor system to verify its identity. You do this by configuring the zone's **TLS verify mode** setting.

If TLS verification is enabled, the neighbor system's FQDN or IP address, as specified in the **Peer address** field of the zone's configuration, is used to verify against the certificate holder's name contained within the X.509 certificate presented by that system. (The name has to be contained in either the Subject Common Name or the Subject Alternative Name attributes of the certificate.) The certificate itself must also be valid and signed by a trusted certificate authority.

Note that for traversal server zones, the FQDN or IP address of the connecting traversal client is not configured, so the required certificate holder's name is specified separately.

If the neighbor system is another VCS, or it is a traversal client / traversal server relationship, the two systems can be configured to authenticate each other's certificates. This is known as mutual authentication and in this case each VCS acts both as client and as a server and therefore you must ensure that each VCS's certificate is valid both as a client and as a server.

See the [Security certificates](#) section for more information on certificate verification and for instructions on uploading the VCS's server certificate and uploading a list of trusted certificate authorities.

Connections to neighbor systems over TCP and TLS

Connections between the VCS and neighbor systems must be configured to use the same SIP transport type, that is they must both be configured to use TLS or both be configured to use TCP.



In software versions prior to X5.1 a connection could be established if one system was configured to use TLS and the other used TCP.

Note that any connection failures due to transport type mismatches are recorded in the Event Log.

SIP authentication trust

If a VCS is configured to use [device authentication](#) it will authenticate incoming SIP registration and INVITE requests. If the VCS then forwards the request on to a neighbor zone such as another VCS, that receiving system will also authenticate the request. In this scenario the message has to be authenticated at every hop.

To simplify this so that a device's credentials only have to be authenticated once (at the first hop), and to reduce the number of SIP messages in your network, you can configure neighbor zones to use the **Authentication trust mode** setting.

Setting a zone's **Authentication trust mode** to **On** means that if the VCS receives an authenticated SIP request from that zone it will trust that authentication and not challenge it again.

If **Authentication trust mode** is **Off** the VCS will always challenge the request even if it has already been authenticated by the sending zone.



Authentication trust only applies when device authentication is enabled.

Note that authenticated SIP requests are identified by the presence of a P-Asserted-Identity field in the SIP message header as defined by [RFC 3325 \[35\]](#).



You are recommended to enable authentication trust only if the neighbor zone is part of a network of trusted SIP servers.

Configuring zones

Common zone configuration options

The zone configuration options depend upon the zone **Type**, however the following options apply to every zone type:

Name

The name acts as a unique identifier, allowing you to distinguish between zones of the same type.

Type

The nature of the specified zone, in relation to the local VCS:

Neighbor: a connection to a neighbor system of the local VCS.

Traversal client: the local VCS is a traversal client of the system being connected to, and there is a firewall between the two.

Traversal server: the local VCS is a traversal server for the system being connected to, and there is a firewall between the two.

ENUM: the zone contains endpoints discoverable by ENUM lookup.

DNS: the zone contains endpoints discoverable by DNS lookup. After a zone has been created, the **Type** cannot be changed.

Hop count

The hop count is the number of times a request will be forwarded to a neighbor gatekeeper or proxy (see the [Hop counts](#) section for more information). This field specifies the hop count to use when sending a search request to this particular zone. If the search request was received from another zone and already has a hop count assigned, the lower of the two values is used.



After creating a zone you must make it a target of at least one of your zone policy search rules. If the zone is not included as a target in your search rules, search requests will not be sent to that zone. See the [Configuring search and zone transform rules](#) section for more information.

The sections that follow describe the specific additional configuration options that apply to each zone **Type**.

Configuring neighbor zones

The following options are available (in addition to the **Name**, **Type** and **Hop count** described in the [Configuring zones](#) section) when configuring a neighbor zone on the VCS. Neighbor zones are used to enable a connection from the local VCS to a neighbor system.



Systems that are configured as cluster peers (formerly known as Alternates) must not be configured as neighbors to each other.

H.323

Mode

Determines whether H.323 calls are allowed to and from the neighbor system.

Port

Specifies the port on the neighbor system used for H.323 calls from the local VCS.

This must be the same port number as that configured on the neighbor system as its H.323 UDP port. If the neighbor is another VCS, this is the port found under the **VCS configuration > Protocols > H.323** in the **Registration UDP Port** field.

SIP

Mode

Determines whether SIP calls are allowed to and from the neighbor system.

Port

Specifies the port on the neighbor system used for SIP connections from the local VCS.

This must be the same port number as that configured on the neighbor system as its SIP TCP, SIP TLS or SIP UDP listening port (depending on which **SIP transport** mode is in use).

Transport

Determines which transport type is used for SIP calls to and from the neighbor system.

The default is **TLS**.

TLS verify mode

Controls whether the VCS performs X.509 certificate checking against the neighbor system when communicating over TLS.

If the neighbor system is another VCS, both systems can verify each other's certificate (known as mutual authentication).

See [TLS certificate verification of neighbor systems](#) for more information.

Authentication trust mode

Controls whether authenticated SIP messages (ones containing a P-Asserted-Identity header) from this zone are trusted without further challenge.

See [SIP authentication trust](#) for more information.

Accept proxied registrations

Controls whether proxied SIP registrations routed through this zone are accepted.

This setting only applies to registration requests for a domain for which the VCS is acting as a Registrar. For requests for other domains the **SIP Registration Proxy Mode** setting applies.

See [Proxying registration requests](#) for more information.

Location

Peer 1 to Peer 6 address

The IP address or FQDN of the neighbor system. If the neighbor is a VCS cluster, this includes all of the peers in the cluster.

See the [Neighboring the local Cisco VCS to another VCS cluster](#) section for more information.

Advanced

See the [Zone configuration: advanced settings](#) section for details on the **Advanced** settings.



Do not use the **Custom** option or configure the individual **Advanced** settings except on the advice of Cisco customer support.

Configuring traversal client zones

The following options are available (in addition to the **Name**, **Type** and **Hop count** described in the [Configuring zones](#) section) when configuring a traversal client zone on the VCS. Traversal client zones are used to enable a connection from the local VCS to a traversal server.

For full details on how traversal client zones and traversal server zones work together to achieve firewall traversal, see the [Firewall traversal](#) section.



An [NTP server](#) must be configured for traversal zones to work.

Authentication username and password

Traversal clients must always authenticate with traversal servers by providing their authentication credentials.

Each traversal client zone must specify an **Authentication username** and **Authentication password** to be used for authentication with the traversal server.



Multiple traversal client zones can be configured on a VCS, each with distinct credentials, to connect to one or more service providers.

Note that the [outbound connection credentials](#) username and password are used for connections to all other (non traversal server) external systems.

H.323

Mode

Determines whether H.323 calls are allowed to and from the traversal server.

Protocol

Determines which of the two firewall traversal protocols (Assent or H.460.18) to use for calls to the traversal server. (See the [H.323 firewall traversal protocols](#) section for more information.)

Port

The port on the traversal server to use for H.323 calls to and from the local VCS.



For firewall traversal to work via H.323, the traversal server must have a traversal server zone configured on it to represent this VCS, using this same port number.

SIP

Mode

Determines whether SIP calls are allowed to and from the traversal server.

Port

The port on the traversal server to use for SIP calls to and from the VCS.

Transport

The transport type to use for SIP calls to and from the traversal server. The default is **TLS**.



For firewall traversal to work via SIP, the traversal server must have a traversal server zone configured on it to represent this VCS, using this same transport type and port number.

TLS verify mode

Controls X.509 certificate checking and mutual authentication between this VCS and the traversal server when communicating over TLS.

See [TLS certificate verification of neighbor systems](#) for more information.

Accept proxied registrations

Controls whether proxied SIP registrations routed through this zone are accepted.

This setting only applies to registration requests for a domain for which the VCS is acting as a Registrar. For requests for other domains the **SIP Registration Proxy Mode** setting applies (see [Proxying registration requests](#)).

Poison mode

Determines if SIP requests sent to systems located via this zone are "poisoned" such that if they are received by this VCS again they will be rejected.

Client settings

Retry interval

Specifies the interval in seconds with which a failed attempt to establish a connection to the traversal server should be retried.

Location

Peer 1 to Peer 6 address

The IP address or FQDN of the traversal server.

If the traversal server is a VCS Expressway cluster, this should include all of its peers. See the [Neighboring the local Cisco VCS to another VCS cluster](#) section for more information.

If the traversal server is a TANDBERG Border Controller, this should include all its Alternates.

Configuring traversal server zones

The following options are available (in addition to the **Name**, **Type** and **Hop count** described in the [Configuring zones](#) section) when configuring a traversal server zone on the VCS Expressway. Traversal server zones are used to enable a connection from the local VCS Expressway to a traversal client.

For full details on how traversal client zones and traversal server zones work together to achieve firewall traversal, see the [Firewall traversal](#) section.



An [NTP server](#) must also be configured in order for traversal zones to work.

Client authentication username

Traversal clients must always authenticate with traversal servers by providing their authentication credentials. The authentication username is the name that the traversal client must provide to the VCS Expressway.

- If the traversal client is a VCS, this must be its **Authentication Username**.
- If the traversal client is a TANDBERG Gatekeeper, this is its **System Name**.

There must also be an entry in the VCS Expressway's local authentication database for the client's authentication username and password. To check the list of entries and add it if necessary, go to the **Local authentication database** page. Either:

- click on the **Add/Edit local authentication database** link
- go to **VCS configuration > Authentication > Local database**

See the [Device authentication](#) section for more information.

H.323

Mode

Determines whether H.323 calls are allowed to and from the traversal client.

Protocol

Determines the protocol (Assent or H.460.18) to use to traverse the firewall/NAT. (See the [H.323 firewall traversal protocols](#) section for more information.)

Port

The port on the local VCS Expressway to use for H.323 calls to and from the traversal client.

H.460.19 demultiplexing mode

Determines whether or not the same two ports are used for media by two or more calls.

On: all calls from the traversal client use the same two ports for media.

Off: each call from the traversal client uses a separate pair of ports for media.

SIP

Mode

Determines whether SIP calls are allowed to and from the traversal client.

Port

The port on the local VCS Expressway to use for SIP calls to and from the traversal client.

Transport

Determines which transport type is used for SIP calls to and from the traversal client. The default is **TLS**.

TLS verify mode and subject name

Controls X.509 certificate checking and mutual authentication between this VCS and the traversal client.

If **TLS verify mode** is enabled, a **TLS verify subject name** must be specified. This is the certificate holder's name to look for in the traversal client's X.509 certificate.

See [TLS certificate verification of neighbor systems](#) for more information.

Accept proxied registrations

Controls whether proxied SIP registrations routed through this zone are accepted.

This setting only applies to registration requests for a domain for which the VCS is acting as a Registrar. For requests for other domains the **SIP Registration Proxy Mode** setting applies (see [Proxying registration requests](#)).

Poison mode

Determines if SIP requests sent to systems located via this zone are "poisoned" such that if they are received by this VCS again they will be rejected.

UDP/TCP probes

UDP retry interval

The frequency (in seconds) with which the client sends a UDP probe to the VCS Expressway if a keep alive confirmation has not been received.

UDP retry count

The number of times the client attempts to send a UDP probe to the VCS Expressway during call setup.

UDP keep alive interval

The interval (in seconds) with which the client sends a UDP probe to the VCS Expressway after a call is established, in order to keep the firewall's NAT bindings open.

TCP retry interval

The interval (in seconds) with which the traversal client sends a TCP probe to the VCS Expressway if a keep alive confirmation has not been received.

TCP retry count

The number of times the client attempts to send a TCP probe to the VCS Expressway during call setup.

TCP keep alive interval

The interval (in seconds) with which the traversal client sends a TCP probe to the VCS Expressway when a call is in place, in order to maintain the firewall's NAT bindings.



The default UDP and TCP probe retry intervals are suitable for most situations. However, if you experience problems with NAT bindings timing out, they may need to be changed.

Configuring ENUM zones

The following options are available (in addition to the **Name**, **Type** and **Hop count** described in the [Configuring zones](#) section) when configuring an ENUM zone. ENUM zones are used to enable ENUM dialing via the local VCS.

Full details of how to use and configure ENUM zones are given in the [ENUM dialing](#) section.

DNS suffix

Specifies the domain to be appended to the transformed E.164 number to create an ENUM domain for which this zone is queried.

H.323 mode

Determines whether H.323 records are looked up for this zone.

SIP mode

Determines whether SIP records are looked up for this zone.

Configuring DNS zones

The following options are available (in addition to the **Name**, **Type** and **Hop count** described in the [Configuring zones](#) section) when configuring a DNS zone. DNS zones are used to enable the local VCS to locate endpoints and other systems by using DNS lookups.

Full details of how to use and configure DNS zones are given in the [Adding and configuring DNS zones](#) section.

H.323 mode

Determines whether H.323 calls are allowed to systems and endpoints located using DNS lookups via this zone.

SIP**Mode**

Determines whether SIP calls are allowed to systems and endpoints located using DNS lookups via this zone.

TLS verify mode

Controls whether the VCS performs X.509 certificate checking against the destination system server returned by the DNS lookup.

This setting only applies if the DNS lookup specifies TLS as required protocol. If TLS is not required then the setting is ignored.

See [TLS certificate verification of neighbor systems](#) for more information.

Advanced

See the [Zone configuration: advanced settings](#) section for details on the **Advanced** settings.



Do not configure the individual **Advanced** settings except on the advice of Cisco customer support.

Zone configuration: advanced settings



The table below describes the **Advanced** and **Custom** zone configuration options. Some of these settings only apply to specific zone types.

You should only use the **Custom** zone profile settings on the advice of Cisco customer support.

Setting	Description	Default	Applicable to
Zone profile	Determines the configuration of the Advanced settings for this zone. The options are: Default: the VCS uses the default values for these settings. Preconfigured profiles: alternatively, choose one of the preconfigured profiles to automatically use the appropriate settings required for connections to that type of system. The options include: • Microsoft Office Communications Server 2007 (see VCS Deployment Guide - VCS Control interworking with Microsoft OCS [24] for full details on how to configure the VCS and OCS to enable the two systems to work together) • Cisco Unified Communications Manager • Nortel Communication Server 1000 • TANDBERG Advanced Media Gateway Custom: lets you configure each Advanced setting individually. These settings are listed in the remainder of this table below.	Default	Neighbor zones DNS zones
Searches are automatically responded to	Determines what happens when the VCS receives a SIP search that originated as an H.323 search. Off: a SIP OPTION or SIP INFO message is sent. On: searches are responded to automatically, without being forwarded. This option should normally be left as the default Off. However, some systems such as Microsoft Office Communications Server (OCS) 2007 do not accept SIP OPTION messages, so for these zones it must be set to On. If you change this to On, you must also configure pattern matches to ensure that only those searches that actually match endpoints in this zone are responded to. If you do not, the search will not continue to other lower-priority zones, and the call will be forwarded to this zone even if it cannot support it.	Off	Neighbor zones DNS zones
Empty INVITE allowed	Determines whether the VCS generates a SIP INVITE message with no SDP to send via this zone. INVITES with no SDP mean that the destination device is asked to initiate the codec selection, and are used when the call has been interworked locally from H.323. On: SIP INVITES with no SDP are generated. Off: SIP INVITES are generated and a pre-configured SDP is inserted before the INVITES are sent. In most cases this option should normally be left as the default On. However, some systems such as Microsoft OCS 2007 do not accept invites with no SDP, so for these zones this should be set to Off.  The settings for the pre-configured SDP are configurable using the xConfiguration Zones Zone [1..1000] DNS Interworking SIP commands. They should only be changed on the advice of Cisco customer support.	On	Neighbor zones DNS zones

Zone configuration: advanced settings

Setting	Description	Default	Applicable to
SIP poison mode	On: SIP requests sent to systems located via this zone are "poisoned" such that if they are received by this VCS again they will be rejected. Off: SIP requests sent out via this zone that are received by this VCS again will not be rejected; they will be processed as normal.	Off	Neighbor zones Traversal client zones Traversal server zones DNS zones
SIP encryption mode	Determines whether or not the VCS allows encrypted SIP calls on this zone. Auto: SIP calls are encrypted if a secure SIP transport (TLS) is used. Off: SIP calls are never encrypted. This option should normally be left as the default Auto. However, this must be set to Off for Microsoft Office Communications Server (OCS) 2007 zones.	Auto	Neighbor zones
SIP SDP attribute line limit mode	Determines whether requests containing SDP sent out to this zone have the length of a=fmtp lines restricted. On: the length is truncated to the maximum length specified by the SIP SDP attribute line limit length setting. Off: the length is not truncated. The SIP SDP attribute line limit option should normally be left as the default of Off. However, some systems such as Microsoft OCS 2007 cannot handle attribute lines longer than 130 characters, so it must be set to On for connections to these systems.	Off	Neighbor zones DNS zones
SIP SDP attribute line limit length	If SIP SDP attribute line limit mode is set to On , sets the maximum line length of a=fmtp SDP lines.	130	Neighbor zones DNS zones
SIP multipart MIME strip mode	Controls whether or not multipart MIME stripping is performed on requests from this zone. This option should normally be left as the default Off. However, it must be set to On for connections to a Microsoft OCS 2007 Release 2 system.	Off	Neighbor zones
SIP UPDATE strip mode	Controls whether or not the VCS strips the UPDATE method from the Allow header of all requests and responses received from this zone. This option should normally be left as the default Off. However, some systems such as Microsoft OCS 2007 do not support the UPDATE method in the Allow header, so for these zones this should be set to On.	Off	Neighbor zones
Interworking SIP search strategy	Determines how the VCS searches for SIP endpoints when interworking an H.323 call. Options: the VCS sends an OPTIONS request. Info: the VCS sends an INFO request. This option should normally be left as the default Options. However, some endpoints such as Microsoft Office Communicator (MOC) clients cannot respond to OPTIONS requests, so this must be set to Info for connections to a Microsoft OCS 2007 system.	Options	Neighbor zones

Zone configuration: advanced settings

Setting	Description	Default	Applicable to
SIP UDP/BFCP filter mode	Determines whether INVITE requests sent to this zone filter out UDP/BFCP. This option may be required to enable interoperability with SIP devices that do not support the UDP/BFCP protocol, so this must be set to On for connections to a Cisco Unified Communications Manager. On: any media line referring to the UDP/BFCP protocol is replaced with TCP/BFCP and disabled. Off: INVITE requests are not modified.	Off	Neighbor zones DNS zones
SIP Duo Video filter mode	Determines whether INVITE requests sent to this zone filter out Duo Video. This option may be required to enable interoperability with SIP devices that do not support Duo Video, so this must be set to On for connections to a Cisco Unified Communications Manager. On: the second video line in any outgoing INVITE request is removed. Off: INVITE requests are not modified.	Off	Neighbor zones DNS zones
SIP record route address type	Controls whether the VCS uses its IP address or host name in the record-route or path headers of outgoing SIP requests to this zone. IP: uses the VCS's IP address. Hostname: uses the VCS's Local host name (if it is blank the IP address is used instead).	IP	Neighbor zones DNS zones
SIP Proxy-Require header strip list	A comma separated list of option tags to search for and remove from Proxy-Require headers in SIP requests received from this zone.	None	Neighbor zones
Include address record	Determines whether, if no NAPTR (SIP) or SRV (SIP and H.323) records have been found for the dialed alias via this zone, the VCS will then query for A and AAAA DNS records before moving on to query lower priority zones. If A and AAAA records exist at the same domain for systems other than those that support SIP or H.323, this may result in the VCS believing the search was successful and forwarding calls to this zone, and the call will fail. On: the VCS queries for A or AAAA records. If any are found, the VCS will not then query any lower priority zones. Off: the VCS will not query for A and AAAA records and instead will continue with the search, querying the remaining lower priority zones.	Off	DNS zones

Zone configuration: pre-configured profile settings

The table below shows the advanced zone configuration option settings that are automatically applied for each of the pre-configured profiles.

Setting	Microsoft Office Communications Server 2007	Cisco Unified Communications Manager	Nortel Communication Server 1000	TANDBERG Advanced Media Gateway
Searches are automatically responded to	Off	Off	Off	Off
Empty INVITE allowed	Off	On	On	On
SIP poison mode	On	Off	Off	Off
SIP encryption mode	Off	On	On	On
SIP SDP attribute line limit mode	On	Off	Off	Off
SIP SDP attribute line limit length	130	130	130	130
SIP multipart MIME strip mode	On	Off	Off	Off
SIP UPDATE strip mode	On	Off	Off	Off
Interworking SIP search strategy	Info	Options	Options	Options
SIP UDP/BFCP filter mode	Off	On	Off	Off
SIP Duo Video filter mode	On	Off	Off	Off
SIP record route address type	Hostname	IP	IP	IP
SIP Proxy-Require header strip list	<blank>	<blank>	"com.nortelnetworks.firewall"	<blank>

Structuring your dial plan

About dial plans

As you start deploying more than one VCS, it is useful to neighbor the systems together so that they can query each other about their registered endpoints. Before you start, you should consider how you will structure your dial plan. This will determine the aliases assigned to the endpoints, and the way in which the VCSs are neighbored together. The solution you choose will depend on the complexity of your system. Some possible options are described in the following sections.

Flat dial plan

The simplest approach is to assign each endpoint a unique alias and divide the endpoint registrations between the VCSs. Each VCS is then configured with all the other VCS as neighbor zones. When one VCS receives a call for an endpoint which is not registered with it, it will send out a Location Request to all the other neighbor VCSs.

While conceptually simple, this sort of flat dial plan does not scale very well. Adding or moving a VCS requires changing the configuration of every VCS, and one call attempt can result in a large number of location requests. This option is therefore most suitable for a deployment with just one or two VCSs plus its peers.

Structured dial plan

An alternative deployment would use a structured dial plan where endpoints are assigned an alias based on the system they are registering with.

If you are using E.164 aliases, each VCS would be assigned an area code. When the VCSs are neighbored together, each neighbor zone would have an associated search rule configured with its corresponding area code as a prefix (a **Mode** of **Alias Pattern Match** and a **Pattern type** of **Prefix**). That neighbor would then only be queried for calls to numbers which begin with its prefix.

In a URI based dial plan, similar behavior may be obtained by configuring search rules for each neighbor with a suffix to match the desired domain name.

It may be desirable to have endpoints register with just the subscriber number -- the last part of the E.164 number. In that case, the search rule could be configured to strip prefixes before sending the query to that zone.

A structured dial plan minimizes the number of queries issued when a call is attempted. However, it still requires a fully connected mesh of all VCSs in your deployment. A hierarchical dial plan can simplify this.

Hierarchical dial plan

In this type of structure one VCS is nominated as the directory for the deployment, and all other VCSs are neighbored with it alone.

The directory VCS is configured with:

- each VCS as a neighbor zone
- search rules for each zone that have a **Mode** of **Alias Pattern Match** and the target VCS's prefix (as with the structured dial plan) as the **Pattern string**

Each VCS is configured with:

- the directory VCS as a neighbor zone
- a search rule with a **Mode** of **Any Alias** and a **Target zone** of the directory VCS

There is no need to neighbor the VCSs with each other. Adding a new VCS now only requires changing configuration on the new VCS and the directory VCS.

However, failure of the directory VCS in this situation could cause significant disruption to communications. Consideration should be given to the use of [Clustering](#) for increased resilience.



For H.323 calls, if **Optimal call routing** is enabled you must ensure that all search rules are configured with a **Source** of **Any**.

If the **Source** is configured to **All zones** (the default), H.323 calls will fail to connect. This is because the H.323 SETUP message, having followed the optimized route established by the original LRQ or ARQ, will appear to the target VCS as coming from an unknown zone. SIP calls, however, are successfully routed if the search rule **Source** is **Any** (because in SIP the search and call setup is combined into one message).

Clustering and peers

This section describes how to set up a cluster of Cisco VCS peers. Clustering is used to increase the capacity of your Cisco VCS deployment and to provide resiliency. The section includes:

- an [overview](#) of clustering
- guidelines for [configuring a cluster](#)
- a list of [configuration that isn't replicated across cluster peers](#)
- a [troubleshooting guide](#) for cluster replication problems
- how [registrations and bandwidth are shared across peers](#)
- how [clustering works with FindMe, Presence and Cisco TMS](#)
- the purpose of the [cluster subzone](#)
- how to [neighbor a local VCS or cluster to a remote VCS cluster](#)



About clustering

A VCS can be part of a cluster of up to six VCSs. Each VCS in the cluster is a peer of every other VCS in the cluster. Clusters are used to:

- increase the capacity of your VCS deployment compared with a single VCS
- provide redundancy in the rare case that a VCS becomes unavailable (for example, due to a network or power outage).

Peers have identical configuration, and share information with each other about their use of bandwidth, registrations, and FindMe users. This allows the cluster to act as one large VCS Local Zone.

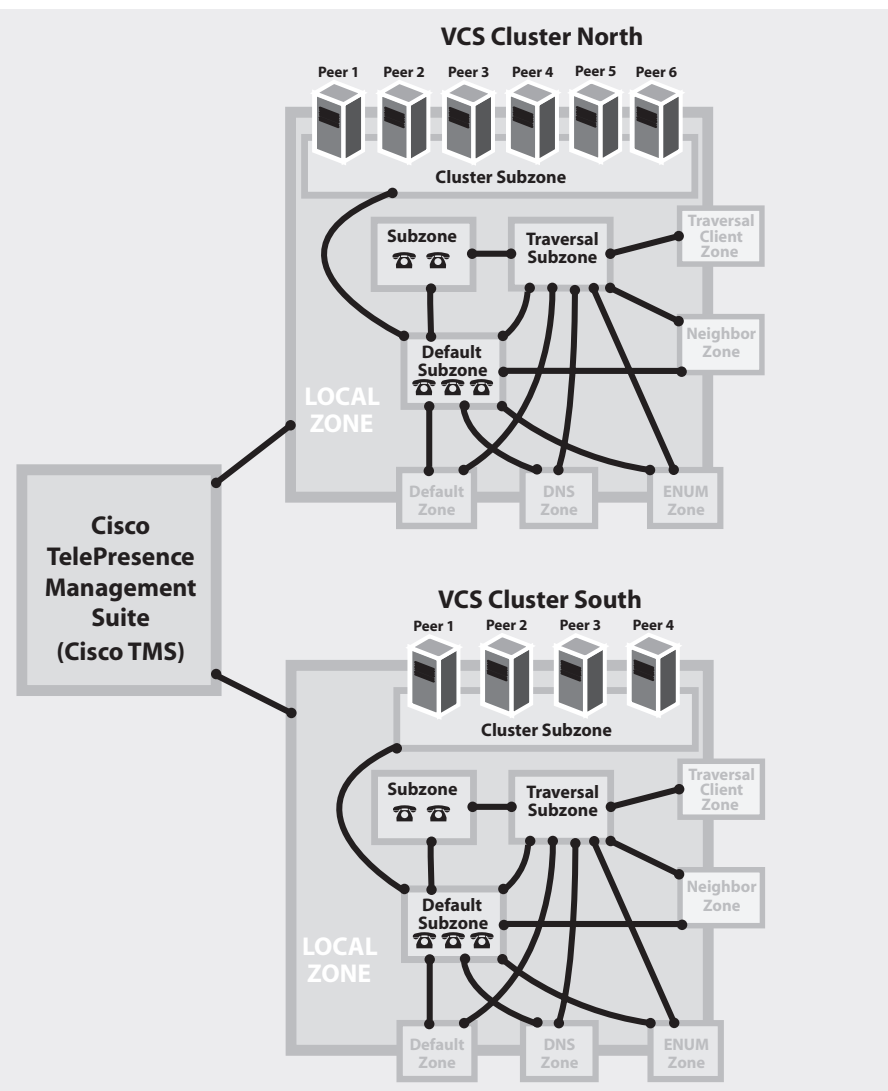
The diagram opposite shows two separate clusters, both being monitored by Cisco TMS.

About the configuration master

All peers in a cluster must be configured identically for subzones, zones, links, pipes, authentication, bandwidth control and call policy. To achieve this, you define a cluster name and nominate one peer as the configuration master. Approximately every minute, all the other peers in the cluster request updated configuration information from the master, and overwrite their existing configuration with these new settings.



"Alternate" is an H.323 term for a system used to provide redundancy to a Primary gatekeeper, and prior to version X3.0 the VCS supported Alternates. From X3.0 onwards, redundancy (along with other features) is provided by clusters of peers, which support both H.323 and SIP and work as equals. However, peers may sometimes be referred to as Alternates. Also note that some versions of Cisco TMS refer to peers as "members".



Configuring clusters

Setting up a cluster

Before creating your cluster, ensure that all the VCSs to be added to the cluster:

- are using [Cisco TMS](#) version 12.5 or later as their external manager
- have the same software version and [option keys](#) installed
- each have a different [system name](#)
- are configured with a working [NTP Server](#).
- each have a different [LAN](#) configuration (a different IPv4 address and a different IPv6 address, where enabled)
- have [H.323](#) enabled (even if all endpoints in the cluster are SIP only, H.323 signaling is used for endpoint location searching and sharing bandwidth usage information with other peers in the cluster)
- have [SSH](#) enabled (data replication between peers uses SSH)
- have [access to the root account via SSH](#) enabled

Configuring the master peer

One of the VCSs in the cluster must act as the master peer for controlling updates to configuration data.

1. Decide which VCS is to be the master and configure it with the settings you want to apply to the entire cluster.
2. Go to the **Clustering** page (**VCS configuration > Clustering**) of the master VCS:
 - a. Enter a **Cluster name** (see [Cluster name](#), right).
 - b. Enter the IP addresses of the master VCS and the other peers into the **Peer IP address** fields.
 - c. Select the **Configuration master** to match the **Peer IP address** number of the master VCS.

Running the clustering script

On each peer in the cluster, log in as **root** and run the cluster configuration script.

A full step-by-step guide on using the clustering script and configuring clusters is available in the [Clustering deployment guide \[27\]](#).

Maintaining a cluster

The **Clustering** page lists the IP addresses of all the peers in the cluster, to which this VCS belongs, and identifies the master peer.

To view the **Clustering** page:

- **VCS configuration > Clustering.**

To view this information from the CLI:

- [xConfiguration Alternates](#)

Setting configuration for the cluster

You should make configuration changes on the master VCS. Any changes made on other peers are not reflected across the cluster, and will be overwritten the next time the master's configuration is replicated across the peers.

The only exceptions to this are:

- some [specific configuration items that are not replicated](#)
- user account details (you can maintain these on any peer — FindMe data uses a different replication mechanism)

You may need to wait up to one minute before changes are updated across all peers in the cluster.

Adding and removing peers from a cluster

After a cluster has been set up you can add new peers to the cluster or remove peers from it. Instructions for this are also contained in the [Clustering deployment guide \[27\]](#).



Systems that are configured as peers must not also be configured as neighbors to each other, and vice versa.



If peers are deployed on different LANs, there must be sufficient connectivity between the networks to ensure a low degree of latency between the peers - a maximum delay of 15ms one way, 30ms round-trip.

Deploying all peers in a cluster on the same LAN means they can be configured with the same routing information such as local domain names and local domain subnet masks.

Cluster name

The **Cluster name** is used to identify one cluster of VCSs from another. Set it to the fully qualified domain name used in SRV records that address this VCS cluster, for example "cluster1.example.com".

See the [Clustering and FindMe](#) section if you are using FindMe.

Changing the master peer

You should only need to change the **Configuration master** when:

- **The original master peer fails:** in this case you must log in to every other VCS in the cluster and change the configuration master on each. Note that if the master fails, the remaining peers will continue to function normally, except they are no longer able to copy their configuration from the master so may become out of sync with each other.
- **You want to take the master VCS unit out of service:** in this case you must log in to the master peer and change the configuration master to another peer. (If you use any other VCS to change the configuration master, this setting, like all other configuration, will be overwritten when that peer next replicates its configuration from the master.)

To change the master peer:

1. Log into the current master VCS and go to the **Clustering** page.
2. Change the **Configuration master** to the peer you want to set as the new master (the numbers match against the **Peer IP address** fields underneath).
3. Click **Save**.

You may need to wait up to one minute before the change is updated across all peers in the cluster.

Monitoring the status of the cluster

The status areas at the bottom of the **Clustering** page show you the current status of the cluster, and the time of the previous and next synchronization.

From here you can also go to the **TMS Agent replication status** page. This shows the current status of the [TMS Agent](#) service and can be used to assist in troubleshooting replication problems.

Which configuration is not replicated?

Most items of configuration are replicated from the master to all peers, with the exceptions listed below.

System name

The [system name](#) is not replicated. It must be different for each peer in the cluster.

Administrator accounts

The password for the default **admin** administrator account is not replicated. Each peer can have a different password.

Any other local administrator accounts and passwords are replicated from the master peer to all other peers.

Option keys

[Option keys](#) are not replicated. Each peer must have an identical set of option keys installed, but you must purchase these separately for each peer in the cluster.

Ethernet speed

The [ethernet speed](#) is not replicated. Each peer may have slightly different requirements for the connection to their ethernet switch.

Conference Factory template

The **Template** used by the [Conference Factory](#) application to route calls to the MCU is not replicated, as it must be unique for each peer.

DNS configuration

[DNS servers](#) are not replicated across peers - each peer can use a different set of DNS servers. However, the [DNS domain name](#) is replicated across peers.

IP configuration

[LAN](#) configuration is not replicated across peers. Each peer must have a different IPv4 address and a different IPv6 address.

[IP gateway](#) configuration is not replicated. Each peer can use a different gateway.

[IP routes](#) (also known as static routes) are not replicated. If these are used, they can be different for each peer.

Note that the [IP protocol](#) is replicated, because each peer must support the same protocol(s).

Logging

The Event Log and Configuration Log on each peer will only report activity for that particular VCS. We recommend that you set up a remote syslog server to which the logs of all peers can be sent. This will allow you to have a global view of activity across all peers in the cluster. See the [About remote logging](#) section for further details.

Certificates

The security certificates and certificate revocation lists (CRLs) used by the VCS must be uploaded individually per peer.



Configuration data that is replicated across peers should not be modified on non-master peers. At best it will result in the changes being overwritten from the master; at worst it will cause cluster replication to fail.

Troubleshooting cluster replication problems

Cluster replication can fail for a variety of reasons. The most common problems are listed below, followed by instructions for resolving the problem:

NTP servers not configured and active on each cluster peer

1. For each peer in the cluster, go to the **System configuration > Time** page.
2. Ensure the peer has a correctly configured and active **NTP server**.

Some peers have a different master peer defined

1. For each peer in the cluster, go to the **VCS configuration > Clustering** page.
2. Ensure each peer identifies the same **Configuration master**.

Cluster configuration script has not been run against each peer

1. For each peer in the cluster, go to the **VCS configuration > Clustering** page.
2. Enter the address of each of peer into the **Peer IP address** fields and configure the **Configuration master**. Ensure each peer identifies the same **Configuration master** peer.
3. Log in to each peer as **root** (by default you can only do this using a serial connection or SSH) and run the cluster configuration script (full details on running this script and configuring clusters is available in the [Clustering deployment guide \[27\]](#)).



Cluster replication warnings can appear briefly while the cluster is initially being set up. These warnings are removed after the data has completed synchronizing and the cluster has stabilized. This takes approximately 3 minutes.

Unable to reach the cluster configuration master peer

The VCS operating as the master peer could be unreachable for many reasons, including:

- network access problems
- VCS unit is powered down
- incorrectly configured IP addresses

"Manual synchronization of configuration is required" warnings are raised on peer VCSs

1. Log in to the peer as **admin** through the CLI (available by default over SSH and through the serial port).
2. Type **xCommand ForceConfigUpdate**.

This will delete the non-master VCS configuration and force it to update its configuration from the master VCS.



Never issue this command on the master VCS, otherwise all VCS configuration for the cluster will be lost.

Sharing registrations across peers

When one VCS in a cluster receives a search request (such as an LRQ, ARQ or an INVITE), it checks its own and its peers' registration databases before responding. This allows all endpoints in the cluster to be treated as if they were registered with a single VCS.



Peers are periodically queried to ensure they are still functioning. To prevent delays during call setup, any non-functioning peers do not receive LRQs.

H.323 registrations

All the peers in a cluster share responsibility for their H.323 endpoint community. When an H.323 endpoint registers with one peer, it receives a registration response which contains a list of Alternate gatekeepers, populated with a randomly ordered list of the IP addresses of all the other peers in that cluster.

If the endpoint loses contact with the initial peer, it will seek to register with one of the other peers. The random ordering of the list of alternate peers ensures that endpoints that can only store a single alternate peer will failover evenly across the cluster.



When using a cluster, you should change the registration **Time to live** on all peers in the cluster from the default 30 minutes to just a few minutes. This setting determines how often endpoints are required to re-register with their VCS, and reducing this to just a few minutes ensures that if one VCS becomes unavailable, the endpoint will quickly failover to one of its peers. To change this setting, go to **VCS configuration > Protocols > H.323 > Gatekeeper > Time to live**.

SIP registrations

Failover re-registration to a peer applies to H.323 re-registrations only.

The SIP standard currently has no direct equivalent, but some SIP UAs including Movit™ v2.0 (or later) clients support similar functionality. If you configure such endpoints with a SIP server address that is an FQDN, and configure this FQDN to resolve to a round-robin DNS record populated with the IP addresses of all the peers in the cluster, then this could allow the endpoint to re-register with another peer if its connection to the original peer was lost.

Sharing bandwidth across peers

When clustering has been configured, all peers share the bandwidth available to the cluster.

Peers must be configured identically for all aspects of bandwidth control including subzones, links and pipes.

Peers share their bandwidth usage information with all other peers in the cluster, so when one peer is consuming part or all of the bandwidth available within or from a particular subzone, or on a particular pipe, this bandwidth will not be available for other peers.

For general information on how the VCS manages bandwidth, see the [Bandwidth control](#) section.

Upgrades and downgrades

The clustering feature was introduced to the VCS in software release X3.0.

Upgrading from versions prior to X3.0

If you are upgrading from VCS software versions prior to X3.0 and want to implement clustering, you must:

1. Remove any existing Alternate configuration.
2. Upgrade all VCSs to be added to the cluster to the latest VCS software version.
3. Follow the steps outlined in the [Configuring clusters](#) section.

Downgrading

See the [Downgrade procedure](#) section for details on restoring system configuration details.

Backup and restore

The [Backup and restore](#) process saves all configuration information for a particular VCS.

You are recommended to regularly backup not just the master peer but all peers in the cluster. This ensures that peer-specific configuration information (see the [Which configuration is not replicated?](#) section) is saved and can be restored individually for each peer.



Do not restore a backup made on one peer to another peer.

Clustering and FindMe

Overview

Clustering supports the use of [FindMe](#). Each peer has its own FindMe database containing all user account information for the cluster. When a user account is created or edited on one peer, that peer shares the information about the changes to all other peers in the cluster, which then update their own FindMe databases accordingly.



There is a limit of 10,000 FindMe user accounts per VCS cluster. Multiple clusters are required if you have more than 10,000 users.

Note that the replication of FindMe database information uses a different mechanism (the [TMS Agent](#)) to that used to replicate configuration information. Configuration information must be changed on the master peer only, but changes to FindMe information can be made on any peer and will be shared with all other peers.

If you are part of a large enterprise with, for example, Cisco TMS managing several VCS clusters, the FindMe database may contain details of users and devices in other VCS clusters. Different clusters are distinguished by their **Cluster name** (see below). You cannot modify the details of accounts that are not managed in your cluster.

Cluster name

You must define a **Cluster name** if you are using FindMe, even if the VCS is not part of a cluster.

If you change the cluster name after creating your user accounts you will have to reconfigure those accounts to associate them with the new cluster name. You can do this by running the *transferfindmeaccounts* script. Instructions for how to do this are contained in the [FindMe Deployment Guide \[29\]](#).

Alternatively, if you try to edit an account that belongs in a different cluster the system gives you an option to **Move this account to local cluster**. Selecting this option updates that particular account so that it now belongs to your local VCS cluster and hence lets you edit that account's details.

See [Configuring clusters](#) for more information on configuring the cluster name.

Clustering and Presence

Clustering supports the use of [Presence](#).

All peers in the cluster must have identical SIP domain, Presence Server and Presence User Agent (PUA) configuration.

If peers in the cluster have the PUA enabled, each peer publishes information about its own local registrations. This information is routed to a Presence Server authoritative for the cluster's domain.

If peers have the Presence Server enabled, the Presence database is replicated across all peers in the cluster.

When [viewing presence status](#) on a peer in a cluster:

- **Publishers** will show all presentities across the cluster for whom presence information is being published.
- **Presentities** will show any presentity for whom a subscription request has been received on the local VCS only.
- **Subscribers** will show each endpoint from which a subscription request has been received on the local VCS only.

Clustering and Cisco TMS

You are recommended to use Cisco TMS when running a cluster of VCSs. For full information, refer to the Cisco TMS documentation.

- Cisco TMS (version 12.5 or later) is mandatory if your cluster is configured to use FindMe or Device Provisioning.

If you were using Cisco TMS to manage a cluster running a version of the VCS software prior to X5, refer to the [VCS Deployment Guide - Creating a Cluster of VCS peers \[27\]](#) for upgrade instructions.



In previous VCS releases, replication between peers was managed by Cisco TMS. From VCS version X4 onwards, replication is managed by the VCSs themselves.

Cluster Subzone

When two or more VCSs are clustered together, a new subzone is created within the cluster's Local Zone. This is the Cluster Subzone (shown in the diagram in the [Clustering overview](#) section). Any calls between two peers in the cluster will briefly pass via this subzone during call setup. The Cluster Subzone is (like the Traversal Subzone) a virtual subzone used for call routing only, and endpoints cannot register to this subzone. After a call has been established between two peers, the Cluster Subzone will no longer appear in the call route and the call will appear as having come from (or being routed to) the Default Subzone.

The two situations in which a call will pass via the Cluster Subzone are:

- Calls between two endpoints registered to different peers in the cluster.
For example, Endpoint A is registered in the Default Subzone to Peer 1. Endpoint B is also registered in the Default Subzone, but to Peer 2. When A calls B, the call route is shown on Peer 1 as **Default Subzone -> Cluster Subzone**, and on Peer 2 as **Cluster Subzone -> Default Subzone**.
- Calls received from outside the cluster by one peer, for an endpoint registered to another peer.
For example, we have a single VCS for the Branch Office, which is neighbored to a cluster of 4 VCSs at the Head Office. A user in the Branch Office calls Endpoint A in the Head Office. Endpoint A is registered in the Default Subzone to Peer 1. The call is received by Peer 2, as it has the lowest resource usage at that moment. Peer 2 then searches for Endpoint A within the cluster's Local Zone, and finds that it is registered to Peer 1. Peer 2 then forwards the call to Peer 1, which forwards it to Endpoint A. In this case, on Peer 2 the call route will be shown as **Branch Office -> Default Subzone -> Cluster Subzone**, and on Peer 1 as **Cluster Subzone -> Default Subzone**.



If **Call routed mode** is set to **Optimal** and the call is H.323, the call will not appear on Peer 2, and on Peer 1 the route will be **Branch Office > DefaultSubzone**.

Neighboring the local Cisco VCS to another VCS cluster

Overview

You can neighbor your local VCS (or VCS cluster) to a remote VCS cluster; this remote cluster could be a neighbor, traversal client, or traversal server to your local VCS. In this case, when a call is received on your local VCS and is passed via the relevant zone to the remote cluster, it will be routed to whichever peer in that neighboring cluster has the lowest resource usage. That peer will then forward the call as appropriate:

- to one of its locally registered endpoints (if the endpoint is registered to that peer)
- to one of its peers (if the endpoint is registered to another peer in that cluster)
- one of its external zones (if the endpoint has been located elsewhere).

When configuring a connection to a remote cluster, you create a single zone and configure it with details of all the peers in the cluster. Adding this information to the zone will ensure that the call is passed to that cluster regardless of the status of the individual peers.

Note that when you are configuring a connection to a remote cluster, you need to enter the IP address of all peers in the remote cluster when the connection is via a **neighbor** or **traversal client** zone. You do not do this for **traversal server** zones, as these connections are not configured by specifying the remote system's IP address.



Systems that are configured as peers must **not** also be configured as neighbors to each other, and vice versa.

Configuration

To neighbor your local VCS (or VCS cluster) to a remote VCS cluster, you create a single zone to represent the cluster and configure it with the details of all the peers in that cluster:

1. On your local VCS (or, if the local VCS is a cluster, on the master peer), [create a zone](#) of the appropriate type. This zone will represent the connection to the cluster.
2. In the **Location** section, enter the IP address or FQDN of each peer in the remote cluster in the **Peer 1** to **Peer 6 address** fields.

To do this using the CLI:

- [Zones Zone \[1..1000\] Neighbor Peer \[1..6\] Address](#)
- [Zones Zone \[1..1000\] TraversalClient Peer \[1..6\] Address](#)



Ideally you should use IP addresses in these fields. If you use FQDNs instead, each FQDN must be different and must resolve to a single IP address for each peer.



The order in which the peers in the remote VCS cluster are listed here does not matter.



Whenever you add an extra VCS to a cluster (to increase capacity or improve redundancy, for example) you will need to modify any VCSs which neighbor to that cluster to let them know about the new cluster peer.

Call processing

This section provides information on the pages that appear under the **Calls, Search rules, Transforms, Call Policy** and **Advanced Media Gateway** sub-menus of the VCS Configuration menu. These pages are used to configure the way in which the Cisco VCS receives and processes calls.

This section includes the following:

- an [overview of how the VCS searches for the destination endpoint](#)
- the different [types of addresses](#) that can be dialed to initiate a call
- how [hop counts](#) affect the search process
- the [searching and transform process](#)
- how to use [Call Policy](#) to manage calls
- routing calls via the [Cisco TelePresence Advanced Media Gateway](#)
- how to set up your network to handle incoming and outgoing calls made via [URI dialing](#) and [ENUM dialing](#)
- [call configuration](#) options
- how to [identify calls](#)
- how to [disconnect calls](#)



Search process

One of the functions of the VCS is to route calls to their appropriate destination, based on the address or alias received from a locally registered endpoint or external zone.

There are a number of steps involved in determining the destination of a call, and some of these steps can involve transforming the alias or redirecting the call to other aliases. It is important to understand the process before setting up your [dial plan](#) so you can avoid circular references, where an alias is transformed from its original format to a different format, and then back to the original alias.

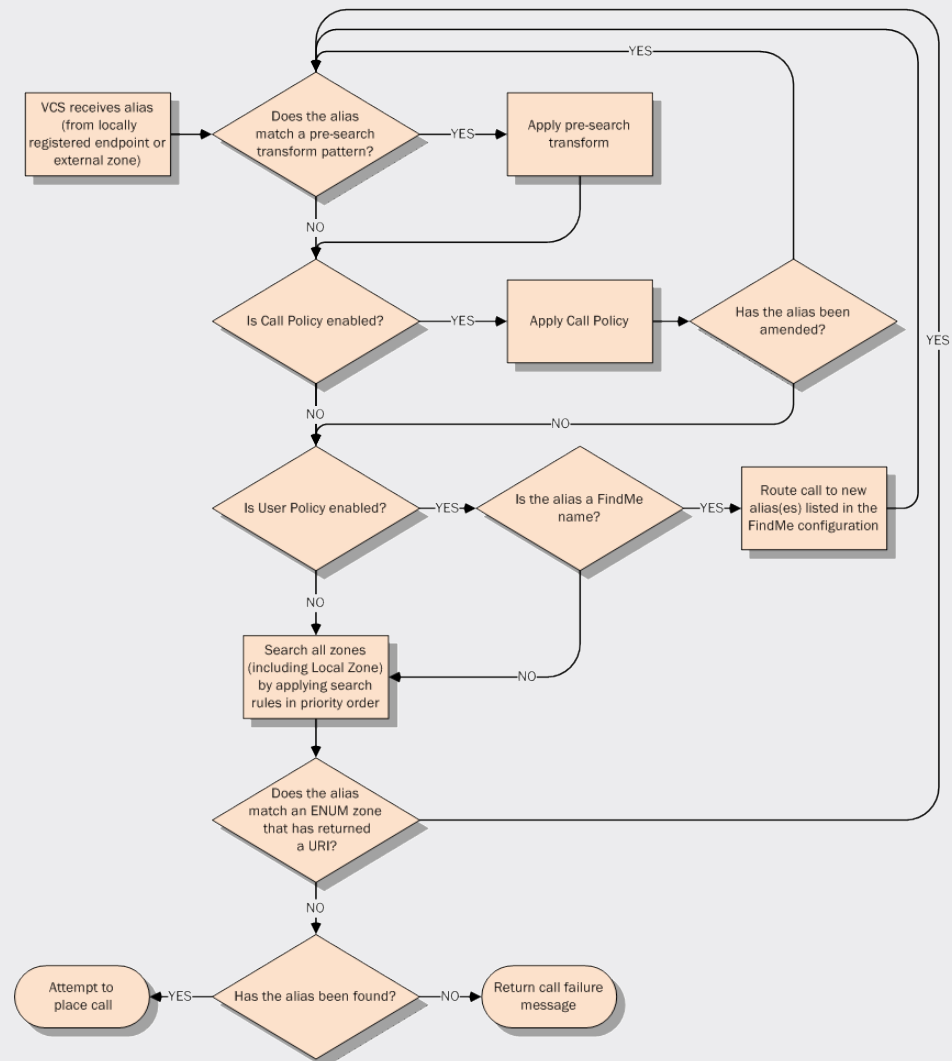


The VCS is able to detect circular references. If it identifies one it will terminate that branch of the search and return a "policy loop detected" error message.

VCS search process

The process followed by the VCS when attempting to locate a destination endpoint is shown in the diagram opposite.

1. The user enters into their endpoint the alias or address of the destination endpoint. This alias or address can be in a number of [different address types](#).
2. The destination address is sent from the caller's endpoint to its local VCS (i.e. the VCS to which it is registered).
3. The VCS applies any [pre-search transforms](#) to the alias.
4. The VCS applies any [Call Policy](#) to the (transformed) alias. If this results in a new alias, the process starts again with the new alias checked against the pre-search transforms.
5. The VCS applies any User Policy (if [FindMe](#) is enabled) to the alias. If the alias is a FindMe ID that resolves to one or more new aliases, the process starts again with all the resulting aliases checked against pre-search transforms and Call Policy.
6. The VCS then applies its [search rules](#) in priority order. At each priority, zones are searched first in the native protocol and then, if the VCS interworking configuration allows, the alternative protocol. If the alias matches an ENUM zone, this may return a URI. If so, the process starts again; the URI is checked against any pre-search transforms, Call Policy and User Policy.
7. If the alias is found within the Local Zone or in one of the external zones, the VCS attempts to place the call to that zone.
8. If the alias is not found, the VCS responds with a message to say that the call has failed.



About the different address types

The destination address that is entered using the caller's endpoint can take a number of different formats, and this will affect the specific process that the VCS follows when attempting to locate the destination endpoint. The address types supported by the VCS are:

- **IP address** e.g. 10.44.10.1 or 3ffe:80ee:3706::10:35
- **H.323 ID** e.g. john.smith or john.smith@example.com (note that an H.323 ID can be in the form of a URI)
- **E.164 alias** e.g. 441189876432 or 6432
- **URI** e.g. john.smith@example.com
- **ENUM** e.g. 441189876432 or 6432

Each of these address types may require some configuration of the VCS in order for them to be supported. The following sections describe the configuration required for each address type.

Dialing by IP address

Dialing by IP address is necessary when the destination endpoint is not registered with any system (such as a VCS, gatekeeper or Border Controller).



If the destination endpoint is registered with one of these systems, it may be possible to call it using its IP address but the call may not succeed if the endpoint is on a private network or behind a firewall. For this reason we recommend that, where possible, calls to registered endpoints are placed using one of the other address types, such as its AOR or H.323 ID. For the same reason we do not recommend that callers outside your network are given IP addresses to use to contact endpoints within your network

The **Calls to Unknown IP addresses** setting determines how the VCS will handle calls made to IP addresses which are not on its local network, or registered with it or one of its neighbors. See the [Calls to unknown IP addresses](#) section for full details on this setting.

Endpoints registered to a Cisco VCS Expressway

Calls made by dialing the IP address of an H.323 endpoint registered directly with a VCS Expressway will be forced to route through the VCS Expressway. The call will therefore be subject to any restrictions configured on that system.

Dialing by H.323 ID or E.164 alias

No special configuration is required to place a call using an H.323 ID or E.164 alias. The VCS follows the usual [search process](#), applying any transforms and then searching the Local Zone and external zones for the alias, according to the search rules.



SIP endpoints will always register using an AOR in the form of a URI. We recommend that H.323 endpoints also register with an H.323 ID in the form of a URI to facilitate interworking.

Dialing by H.323 or SIP URI

When a user places a call using URI dialing, they will typically dial **name@example.com**.

If the destination endpoint is locally registered or registered to a neighbor system, no special configuration is required for the call to be placed. The VCS follows the usual [search process](#), applying any transforms and then searching the Local Zone and external zones for the alias, according to the search rules.

If the destination endpoint is not locally registered, URI dialing may make use of DNS to locate the destination endpoint. To support URI dialing via DNS, you must configure the VCS with at least one DNS server and at least one DNS zone.

Full instructions on how to configure the VCS to support URI dialing via DNS (both outbound and inbound) are given in the [URI dialing](#) section.

Dialing by ENUM

ENUM dialing allows an endpoint to be contacted by a caller dialing an E.164 number - a telephone number - even if that endpoint has registered using a different format of alias. The E.164 number is converted into a URI by the DNS system, and the rules for URI dialing are then followed to place the call.

The ENUM dialing facility allows you to retain the flexibility of URI dialing whilst having the simplicity of being called using just a number - particularly important if any of your callers are restricted to dialing using a numeric keypad.

In order to support ENUM dialing on the VCS you must configure it with at least one DNS server and the appropriate ENUM zone(s).

Full instructions on how to configure the VCS to support ENUM dialing (both outbound and inbound) are given in the [ENUM dialing](#) section.

About hop counts

Each search request is assigned a hop count value by the system that initiates the search. Every time the request is forwarded to another neighbor gatekeeper or proxy, the hop count value is decreased by a value of 1. When the hop count reaches 0, the request will not be forwarded on any further and the search will fail.

For search requests initiated by the local VCS, the hop count assigned to the request is configurable on a zone-by-zone basis. The zone's hop count applies to all search requests originating from the local VCS that are sent to that zone.

Search requests received from another zone will already have a hop count assigned. When the request is subsequently forwarded on to a neighbor zone, the lower of the two values (the original hop count or the hop count configured for that zone) is used.

For H.323, the hop count only applies to search requests. For SIP, the hop count applies to all requests sent to a zone, affecting the Max-Forwards field in the request.

The hop count value can be between **1** and **255**.

The default is **15**.



If your hop counts are set higher than necessary, you may risk introducing loops into your network. In these situations a search request will be sent around the network until the hop count reaches 0, consuming resources unnecessarily. This can be prevented by setting the [Call loop detection mode](#) to **On**.



When dialing by URI or ENUM, the hop count used is that for the associated DNS or ENUM zone via which the destination endpoint (or intermediary SIP proxy or gatekeeper) was found.

Configuring hop counts

Hop counts are configured on a zone basis. To configure the hop count for a zone:

1. Go to the **Zones** page.
VCS configuration > Zones
2. Click on the name of the zone you want to configure.
You are taken to the **Edit zone** page.
3. In the **Configuration** section, in the **Hop Count** field, enter the hop count value you want to use for this zone.

To configure the hop count using the CLI:

- [xConfiguration Zones Zone \[1..1000\] HopCount](#)



For full details on other zone options, see the [Zone configuration](#) section.

Overview of searches and transforms

About searches

One of the VCS's functions is to process incoming requests to search for a particular alias. These search requests are received from:

- locally registered endpoints
- neighboring systems, including neighbors, traversal clients and traversal servers
- endpoints on the public internet

You can configure how the VCS searches its Local Zone and external zones by setting up search rules that let you:

- define alias, IP address and pattern matches to filter searches to specific zones
- define the order (priority) in which the search rules are applied and stop applying any lower-priority search rules once a match is found; this lets you reduce the potential number of search requests sent out, and speed up the search process
- set up different rules according to the source of the query (such as the Local Zone, or any known zone)
- apply zone transforms to modify an alias

You can configure up to 2000 search rules.



The VCS uses the protocol (SIP or H.323) of the incoming call when searching a zone for a given alias. If the search is unsuccessful the VCS may then search the same zone again using the alternative protocol, depending on where the search came from and the **Interworking mode (VCS configuration > Protocols > Interworking)**. If the request has come from a neighboring system and **Interworking mode** is set to **RegisteredOnly**, the VCS searches the Local Zone using both protocols, and all other zones using the native protocol only (because it will interwork the call only if one of the endpoints is locally registered). If **Interworking mode** is set to **On**, or the request has come from a locally registered endpoint, the VCS searches the Local Zone and all external zones using both protocols.

About transforms

The VCS lets you transform the alias in a search request if it matches certain criteria. This transformation can be applied to the alias at two points in the search process: as a pre-search transform, and as a zone transform.

Pre-search transform

Pre-search transforms are applied before any Call Policy, User Policy or search rules are applied (see the [Pre-search transforms](#) section for more details).

Zone transform

Zone transforms are applied after Call Policy, User Policy and the search rules have been applied, and let you change the alias being searched for before a search request is sent to the Local Zone or out to an external zone (see the [Zone searching and transform process](#), right). Zone transforms are specified as a part of the [search rules](#) configuration.

You can transform an alias by removing or replacing its prefix, suffix, or the entire string, and by the use of regular expressions.



Transforms support the use of Regular Expressions in both the **Pattern String** and **Replace String** fields. See the [Regular expression reference](#) Appendix for more information.



Multiple search rules can refer to the same target zone. This means that you can specify different sets of search criteria and zone transforms for each zone.

For full information about configuring search rules, see the [Configuring search and zone transform rules](#) section.

Zone searching and transform process

Zone searching takes place after all pre-search transforms, Call Policy and User Policy have been applied. The search rules and zone transform process is as follows:

1. The VCS applies the search rules in priority order (all rules with a priority of 1 are processed first, then priority 2 and so on) to see if the given alias matches the rules criteria based on the **Source** of the query and the rule **Mode**.
2. If the match is successful, any associated zone transform (where the **Mode** is **Alias Pattern Match** and the **Pattern Behavior** is **Replace**) is applied to the alias.
3. The search rule's **Target zone** is queried (with the revised alias if a zone transform has been applied) using the same protocol (SIP or H.323) as the incoming call request. Note that if there are many successful matches for multiple search rules at the same priority level, every applicable **Target zone** is queried.
 - If the alias is found, the call is forwarded to that zone. If the alias is found by more than one zone, the call is forwarded to the zone that responds first.
 - If the alias is not found using the native protocol, the query is repeated using the interworked protocol, depending on the **Interworking mode**.
4. If the alias is not found, the search rules with the next highest priority are applied (go back to step 1) until:
 - the alias is found, or
 - all target zones associated with search rules that meet the specified criteria have been queried, or
 - a search rule with a successful match has an **On successful match** setting of **Stop searching**



Note the difference between a successful match (where the alias matches the search rule criteria) and an alias being found (where the query sent to the target zone is successful). The **Stop searching** option provides better control over the network's signaling infrastructure. For example, if searches for a particular domain should always be routed to a specific zone this option lets you make the search process more efficient and stop the VCS from searching any other zones unnecessarily.

Pre-search transforms

About pre-search transforms

The pre-search transform function allows you to modify the alias in an incoming search request. The transformation is applied by the VCS before any searches take place, either locally or to external zones. It applies to all incoming search requests received from locally registered endpoints, neighbor, traversal client and traversal server zones, and endpoints on the public internet. It does not apply to requests received from peers (which are configured identically and therefore will have already applied the same transform).

Each pre-search transform defines a string against which an alias is compared, and the changes to make to the alias if it matches that string.

After the alias has been transformed, it remains changed and all further call processing is applied to the new alias.



Pre-search transforms are not applied to GRQ or RRQ messages received from endpoints registering with the VCS; endpoints will be registered with the alias(es) as presented in these messages.



Pre-search transforms are applied prior to any possible CPL modification and Zone transforms.



All peers in a cluster should be configured identically, including any pre-search transforms. A VCS in a cluster treats search requests from any of its peers as having come from its own Local Zone, and does not re-apply any pre-search transforms on receipt of the request.

Pre-search transform process

Up to 100 pre-search transforms can be configured. Each transform must have a unique priority number between **1** and **65534**.

Every incoming alias is compared with each transform in order of priority, starting with that closest to **1**. If and when a match is made, the transform is applied to the alias and no further pre-search checks and transformations of the new alias will take place. The new alias is then used for the remainder of the [search process](#).



Further transforms of the alias may take place during the remainder of the search process. This may be as a result of [Call Policy](#) (also known as Administrator Policy) or User Policy (if [FindMe](#) is enabled). If this is the case, the pre-search transforms are re-applied to the new alias. Refer to the [search process](#) diagram for more information.



If you add a new pre-search transform that has the same priority as an existing transform, all transforms with a lower priority (those with a larger numerical value) will have their priority incremented by one, and the new transform will be added with the specified priority. However, if there are not enough “slots” left to move all the priorities down, you will get an error message.

Pre-search transforms (continued)

Configuring pre-search transforms

The **Transforms** page lists all the pre-search transforms currently configured on the VCS. It is used to create, edit, delete, enable and disable transforms.

To go to the **Transforms** page:

- **VCS configuration > Transforms.**

To configure pre-search transforms using the CLI:

- [xConfiguration Transform \[1..100\]](#)

Click on the transform you want to configure (or click **New** to create a new transform, or click **Delete** to remove a transform).



If you hover your mouse pointer over a transform, the transform description (if one has been defined) appears as a tooltip.

Enabling and disabling transforms

When you are making or testing configuration changes to your transforms, you may want to temporarily enable or disable certain transforms. You can do this by selecting the transform's check box and clicking **Enable** or **Disable** as appropriate. Any disabled transforms still appear in the transforms list but are ignored by the VCS when processing search requests.

Configuration options

The configurable options are:

Priority

Assigns a priority to this transform. Priority can be from **1** to **65534**, with 1 being the highest priority. Transforms are applied in order of priority, and the priority must be unique for each transform.

Description

An optional free-form description of the transform.

Pattern type

The way in which the **Pattern string** must match the alias. Options are:

- Exact:** the string must match the alias character for character.
- Prefix:** the string must appear at the beginning of the alias.
- Suffix:** the string must appear at the end of the alias.
- Regex:** treats the string as a regular expression.

Pattern string

Specifies the pattern against which the alias is compared.

Pattern behavior

Specifies how the matched part of the alias is modified. Options are:

Strip: the matching prefix or suffix is removed.

Replace: the matching part of the alias is substituted with the text in the **Replace string**.

Add Prefix: prepends the **Additional text** to the alias.

Add Suffix: appends the **Additional text** to the alias.

Replace string

(Applies if **Pattern Behavior** is **Replace**.)

The string to be used as a substitution for the part of the alias that matched the pattern.

Additional text

(Applies if **Pattern Behavior** is **Add Prefix** or **Add Suffix**.)

The string to add as a prefix or suffix.

State

Indicates if the transform is enabled or not.



Pre-search transforms support the use of Regular Expressions in both the **Pattern String** and **Replace String** fields. See the [Regular expression reference](#) Appendix for more information.



You can test whether a pattern will match a particular alias and be transformed in the expected way by using the [Check pattern](#) tool (**Maintenance > Tools > Check pattern**).



Note that the transforms also apply to any Publication, Subscription or Notify URIs handled by the [Presence Services](#).

Search configuration

Calls to unknown IP addresses

Although the VCS supports dialing by IP address, it is sometimes undesirable for a VCS to be allowed to place a call directly to an IP address that is not local. Instead, you may want a neighbor to place the call on behalf of the VCS, or not allow such calls at all.

The VCS lets you configure how it behaves when receiving a call for an IP address that is not local through the **Calls to Unknown IP addresses** setting on the **Search rules configuration** page (**VCS configuration > Search rules > Configuration**).

To configure this setting from the CLI:

- [xConfiguration Call Services CallsToUnknownIPAddresses](#)

About unknown IP addresses

The VCS considers an IP address to be "known" if it either:

- is the IP address of a locally registered endpoint
- falls within the IP address range of one of the subzone membership rules configured on the VCS

Calls to these IP addresses are not affected by the **Calls to Unknown IP addresses** setting - the VCS will always attempt to place the call (providing there is a search rule for **Any IP Address** against the Local Zone). All other IP addresses are considered to be "unknown" and are handled by the VCS according to its **Calls to Unknown IP addresses** setting.

About unregistered endpoints

An unregistered endpoint is any device that is not registered with an H.323 gatekeeper or SIP registrar (e.g. VCS, gatekeeper or TANDBERG Border Controller). Although most calls are made between endpoints that are registered with such systems, it is sometimes necessary to place a call to an unregistered endpoint.

There are two ways to call to an unregistered endpoint:

- by dialing its URI (this requires that the local VCS is configured to support URI dialing, and a DNS record exists for that URI that resolves to the unregistered endpoint's IP address)
- by dialing its IP address

Calls to Unknown IP addresses settings

The options for the **Calls to Unknown IP addresses** setting are:

Direct: the VCS attempts to place the call directly to the unknown IP address without querying any neighbors.

Indirect: the VCS forwards the search request to its neighbors in accordance with its normal search process, meaning any zones that are the target of search rules with an **Any IP Address** mode. If a match is found and the neighbor's configuration allows it to connect a call to that IP address, the VCS will pass the call to that neighbor for completion.

Off: the VCS will not attempt to place the call, either directly or to any of its neighbors.



This setting applies to the call's destination address prior to any zone transforms, but after any pre-search transforms, Call Policy or User Policy rules have been applied.



In addition to controlling calls, this setting also determines the behavior of provisioning and presence messages to SIP devices, as these messages are routed to IP addresses.

Recommended configuration for firewall traversal

When the VCS Expressway is neighbored with a VCS Control for firewall traversal, you should typically set **Calls to unknown IP addresses** to **Indirect** on the VCS Control and **Direct** on the VCS Expressway. When a caller inside the firewall attempts to place a call to an IP address outside the firewall, it will be routed as follows:

1. The call will go from the endpoint to the VCS Control with which it is registered.
2. As the IP address being called is not registered to that VCS, and its **Calls to unknown IP addresses** setting is **Indirect**, the VCS will not place the call directly. Instead, it will query its neighbor VCS Expressway to see if that system is able to place the call on the VCS Control's behalf.
3. The VCS Expressway receives the call and because its **Calls to unknown IP addresses** setting is **Direct**, it will make the call directly to the called IP address.

Fallback alias

The VCS could receive a call that is destined for it but which does not specify an alias. This could be because:

- the caller has dialed the IP address of the VCS directly
- the caller has dialed a domain name belonging to the VCS (either one of its configured SIP domains, or any domain that has an SRV record that points at the IP address of the VCS), without giving an alias as a prefix

Normally such calls would be disconnected. However, the VCS allows you to specify an alias to which all such calls should be routed. This alias is known as the **fallback alias** and is configured on the **Search rules configuration** page (**VCS configuration > Search rules > Configuration**).

To configure this setting using the CLI:

- [xConfiguration Call Services Fallback Alias](#)



If no fallback alias is configured, calls that do not specify an alias will be disconnected.



Some endpoints do not allow users to enter an alias and an IP address to which the call should be placed.

Example usage

You may want to configure your fallback alias to be that of your receptionist, so that all calls that do not specify an alias are still answered personally and can then be redirected appropriately.

For example, Example Inc. has the domain of *example.com*. The endpoint at reception has the alias *reception@example.com*.

They configure their VCS with a fallback alias of *reception@example.com*. This means that any calls made directly to *example.com* (i.e. without being prefixed by an alias), are forwarded to *reception@example.com*, where the receptionist answers the call and directs it appropriately.

Zone searching and zone transforms

Configuring search and zone transform rules

The **Search rules** page lists all the existing search rules. It is used to add, edit, delete, enable and disable rules.

Each rule specifies the criteria used to match an alias in the source query before sending the query on to a target zone. The rule also specifies any zone transforms to apply to the alias.

To go to the **Search rules** page:

- **VCS configuration > Search rules > Rules.**

To configure search rules using the CLI:

- [xConfiguration Zones Policy SearchRules](#)



Note that you can click on a column heading to sort the list, for example by **Target zone** or **Priority**. If you hover your mouse pointer over a search rule, the rule description (if one has been defined) appears as a tooltip.

Click on the rule you want to configure (or click **New** to create a new rule, or click **Delete** to remove a rule).

Enabling and disabling search rules

When you are making or testing configuration changes to your search rules, you may want to temporarily enable or disable certain rules. You can do this by selecting the rule's check box and clicking **Enable** or **Disable** as appropriate. Any disabled rules still appear in the search rules list but are ignored by the VCS when processing search requests.

Configuration options

The configurable options are:

Rule name

A descriptive name for the search rule.

Description

An optional free-form description of the search rule.

Priority

The order in the search process that this rule is applied, when compared to the priority of the other search rules. All Priority 1 search rules are applied first, followed by all Priority 2 search rules, and so on. More than one rule can be assigned the same priority, in which case any matching target zones are queried simultaneously.

Different default priorities are assigned depending on the type of the **Target zone**:

- **Local Zone:** 50
- **Neighbor, traversal client or traversal server zone:** 100
- **ENUM or DNS zone:** 150

This default configuration means that the Local Zone is searched first for all aliases. If the alias is not found locally, all neighbor, traversal client and traversal server zones are searched, and if they cannot locate the alias the request is sent to any DNS and ENUM zones.

Source

Specifies the sources of the requests for which this rule applies.

Any: locally registered devices, neighbor or traversal zones, and any non-registered devices.

All zones: locally registered devices plus neighbor or traversal zones.

Local Zone: locally registered devices only.

Mode

Determines whether a query is sent to the **Target zone**, based on the given alias.

Alias Pattern Match: queries the zone only if the alias matches the corresponding **Pattern type** and **Pattern string**.

Any Alias: queries the zone for any alias (but not IP address).

Any IP Address: queries the zone for any given IP address (but not alias).

Zone searching and zone transforms (continued)

Pattern type

How the pattern string must match the alias for the rule to be applied. (Applies only if the **Mode** is **Alias Pattern Match**.)

Exact: the entire string must exactly match the alias character for character.

Prefix: the string must appear at the beginning of the alias.

Suffix: the string must appear at the end of the alias.

Regex: treats the string as a regular expression.

Pattern string

The pattern against which the alias is compared. (Applies only if the **Mode** is **Alias Pattern Match**.)

Pattern behavior

Determines whether the matched part of the alias is modified before being sent to the target zone. (Applies only if the **Mode** is **Alias Pattern Match**.)

Leave: the alias is not modified.

Strip: the matching prefix or suffix is removed from the alias.

Replace: the matching part of the alias is substituted with the text in the **Replace string**. The **Target zone** is then queried using the revised alias. Note that if you want to transform the alias before applying search rules you must use [pre-search transforms](#).

Replace string

The string to substitute for the part of the alias that matches the pattern. (Applies only if the **Pattern behavior** is **Replace**.)

On successful match

Controls the ongoing search behavior if the alias matches the search rule.

Continue: continue applying the remaining search rules (in priority order) until the endpoint identified by the alias is found.

Stop: do not apply any more search rules, even if the endpoint identified by the alias is not found in the target zone.

Note that if **Stop** is selected, any rules with the same priority level as this rule are still applied.

Target zone

The zone to query if the alias matches the search rule.

State

Indicates if the search rule is enabled or not.



You can test whether a pattern matches a particular alias and is transformed in the expected way by using the [Check pattern](#) tool.



You can test whether the VCS can find an endpoint identified by a given alias, without actually placing a call to that endpoint by using the [Locate](#) tool.

Examples

Stripping @domain for dialing to H.323 numbers

SIP endpoints can only make calls in the form of URIs - for example name@domain. If the caller does not specify a domain when placing the call, the SIP endpoint automatically appends its own domain to the number that is dialed. So if you dial 123 from a SIP endpoint, the search will be placed for 123@domain. If the H.323 endpoint being dialed is registered as 123, the VCS will be unable to locate the alias 123@domain and the call will fail.

If you have a deployment that includes both SIP and H.323 endpoints that register using a number, you will need to set up the following pre-search and zone transforms. This will let users place calls from SIP and H.323 endpoints to H.323 endpoints registered using their H.323 E164 number only.

Explanation

The pre-search transform (example, below) takes any number-only dial string (such as 123) and appends the domain used in endpoint AORs and URIs in your deployment. This ensures that calls made by SIP and H.323 endpoints result in the same URI.

Search rules (right) for the Local Zone then ensure that both the E164 number and full URI are searched for, so that endpoints can still be reached whether they have registered with an H.323 number (123) or a full URI (123@domain):

- The first search rule takes any aliases in the format *number@domain* and transforms them into the format *number*.
- To ensure that any endpoints that have actually registered with an alias in the format *number@domain* can also still be reached, a second search rule must be configured at a lower priority that places calls to *number@domain* without transforming the alias.

Configuration

Rule name: * Dialing H.323 numbers *i*

Description: Transform aliases in format number@domain to number *i*

Priority: * 50 *i*

Source: Any *i*

Mode: AliasPatternMatch *i*

Pattern type: Regex *i*

Pattern string: * (\\d+)@domain *i*

Pattern behavior: Replace *i*

Replace string: \\1 *i*

On successful match: Continue *i*

Target zone: LocalZone *i*

State: Enabled *i*

Configuration

Rule name: * Dialing H.323 numbers with domain *i*

Description: Place calls to number@domain with no alias transform *i*

Priority: * 60 *i*

Source: Any *i*

Mode: AliasPatternMatch *i*

Pattern type: Regex *i*

Pattern string: * (\\d+)@domain *i*

Pattern behavior: Leave *i*

On successful match: Continue *i*

Target zone: LocalZone *i*

State: Enabled *i*

Create transform

You are here: [VCS configuration](#) > [Transforms](#) > Create transform

Configuration

Priority: 1 *i*

Description: Take any number-only dial string and append @domain *i*

Pattern type: Regex *i*

Pattern string: * (\\d+) *i*

Pattern behavior: Replace *i*

Replace string: \\1@domain *i*

State: Enabled *i*

Examples

Transforms for alphanumeric H.323 ID dial strings

This example builds on the example from the previous page. The previous example caters for number-only dial strings, however H.323 IDs do not have to be purely numeric; they can contain alphanumeric (letters and digits) characters.

This example follows the same model as the previous example — a pre-search transform and two search rules to ensure that endpoints can be reached whether they have registered with an H.323 ID or a full URI — but uses a different regex (regular expression) that supports alphanumeric characters.

Explanation

The pre-search transform (example, below) takes any alphanumeric dial string (such as *123abc*) and appends the domain used in your deployment to ensure that calls made by SIP and H.323 endpoints result in the same URI.

Search rules (right) for the Local Zone then ensure that both the E164 ID and full URI are searched for, so that endpoints can still be reached whether they have registered with an H.323 ID (*123abc*) or a full URI (*123abc@domain*):

- The first search rule takes any aliases in the format *string@domain* and transforms them into the format *string*.
- To ensure that any endpoints that have actually registered with an alias in the format *string@domain* can also still be reached, a second search rule must be configured at a lower priority that places calls to *string@domain* without transforming the alias.

Configuration

Rule name	* Dialing H.323 strings
Description	Transform aliases in the format string@domain to string
Priority	* 40
Source	Any
Mode	AliasPatternMatch
Pattern type	Regex
Pattern string	* (.+)@domain
Pattern behavior	Replace
Replace string	\$1
On successful match	Continue
Target zone	LocalZone
State	Enabled

Configuration

Rule name	* Dialing H.323 strings with domain
Description	Place calls to string@domain without alias transform
Priority	* 50
Source	Any
Mode	AliasPatternMatch
Pattern type	Regex
Pattern string	* (.+)@domain
Pattern behavior	Leave
On successful match	Continue
Target zone	LocalZone
State	Enabled

Create transform

You are here: [VCS configuration](#) > [Transforms](#) > Create transform

Configuration

Priority	* 1
Description	Append @domain to any alphanumeric dial string
Pattern type	Regex
Pattern string	* ([^@]*)
Pattern behavior	Replace
Replace string	\$1@domain
State	Enabled

Examples

Combining match types and priorities

By using the **Any Alias** and **Alias Pattern Match** modes when defining search rules, and applying the same or different priorities to each rule, you will have a great deal of flexibility in determining if and when the target zone is queried and whether any transforms are applied. Some example configurations are given here.



The **Any Alias** mode does not support alias transforms. If you want to always query a zone using a different alias to that received, you need to use a mode of **Alias Pattern Match** in combination with a regular expression.

Always query a zone with original alias (no transforms)

To configure a zone so that it is always sent search requests using the original alias, set up a search rule for that zone with a mode of **Any Alias**.

Configuration

Rule name	* Always query with original alias	i
Description	Send search requests using the original alias	i
Priority	* 100	i
Source	Any	i
Mode	AnyAlias	i
On successful match	Continue	i
Target zone	Head office	i
State	Enabled	i

Configuring a zone for incoming calls only

To configure a zone so that it is never sent an alias search request (for example if you only want to receive incoming calls from this zone), do not define any search rules that have that zone as its target. In this scenario, when viewing the zone, you can ignore the warning that no search rules exist.

Edit zone

You are here: [VCS configuration](#) > [Zones](#) > Edit zone

Search rules not configured: This zone does not appear as the target in any zone search rules. You can configure which zones are searched and in what order on the [Search rules](#) page.

Configuration

Name	* One way zone	i
Type	Neighbor	
Hop count	* 15	i

Examples

Filter queries to a zone without transforming

It is possible to filter the search requests sent to a zone so that it is only queried for aliases that match certain criteria.

For example, assume all endpoints in your regional sales office are registered to their local VCS with a suffix of @sales.example.com.

In this situation, it makes sense for your Head Office VCS to query the Sales Office VCS only when it receives a search request for an alias with a suffix of @sales.example.com. Sending any other search requests to this particular VCS would take up resources unnecessarily.

To achieve this, on your Head Office VCS create a zone to represent the Sales Office VCS, and set up an associated search rule with a **Mode** of **Alias Pattern Match** mode and a **Pattern type** of **Suffix**.

It would also be wasteful of resources to send search requests for aliases that match this pattern to any other zone (there may be other lower priority search rules defined that would also apply to these aliases). In which case setting **On successful match** to **Stop** means that the VCS will not apply any further (lower priority) search rules.

Allowing calls to IP addresses only if they come from known zones

In addition to aliases, calls can be made to specified IP addresses. To pass on such calls to the appropriate target zones you must set up search rules with a mode of **Any IP Address**.

To provide extra security you can set the rule's **Source** option to **All Zones**. This means that the query is only sent to the target zone if it originated from any configured zone or the Local Zone.

Configuration

Rule name	* Regional sales office	i
Description	Calls to aliases with a suffix of @sales.example.com	i
Priority	* 100	i
Source	Any	i
Mode	AliasPatternMatch	i
Pattern type	Suffix	i
Pattern string	* @sales.example.com	i
Pattern behavior	Leave	i
On successful match	Stop	i
Target zone	Sales office	i
State	Enabled	i

Configuration

Rule name	* IP addresses from known zones	i
Description	Allow calls to IP addresses only if from a known zone	i
Priority	* 100	i
Source	AllZones	i
Mode	AnyIPAddress	i
On successful match	Continue	i
Target zone	Overseas office	i
State	Enabled	i

Examples

Query a zone for original and transformed alias

You may want to query a zone for the original alias at the same time as you query it for a transformed alias. To do this, configure one search rule (example top, right) with a mode of **Any Alias**, and a second search rule (example bottom, right) with a mode of **Alias Pattern Match** along with details of the transform to be applied. Both searches must be given the same **Priority** level.

For example, you may want to query a neighbor zone for both a full URI and just the name (the URI with the domain removed).

To achieve this, on your local VCS configure the two search rules that target the zone representing the neighbor VCS as shown.

Configuration

Rule name	* Overseas office - original alias	i
Description	Query overseas office with the original alias	i
Priority	* 100	i
Source	Any	i
Mode	AnyAlias	i
On successful match	Continue	i
Target zone	Overseas office	i
State	Enabled	i

Configuration

Rule name	* Overseas office - strip domain	i
Description	Query overseas office with domain removed	i
Priority	* 100	i
Source	Any	i
Mode	AliasPatternMatch	i
Pattern type	Suffix	i
Pattern string	* @example.com	i
Pattern behavior	Strip	i
On successful match	Continue	i
Target zone	Overseas office	i
State	Enabled	i

Examples

Query a zone for two or more transformed aliases

Zones are queried in order of priority of the search rules configured against them.

It is possible to configure multiple search rules for the same zone each with, for example, the same **Priority** and an identical **Pattern String** to be matched, but with a different replacement patterns.

In this situation, the VCS queries that zone for each of the new aliases simultaneously. (Any duplicate aliases produced by the transforms are removed prior to the search requests being sent out.)

If any of the new aliases are found by that zone, the call is forwarded to the zone. It is then up to the controlling system to determine the alias to which the call will be forwarded.

The example search rules shown here both look for aliases with a suffix of *example.com*. The top example then replaces the suffix with *example.co.uk* and the bottom example replaces it with *example.net*.

Configuration

Rule name

★ Transform to example.co.uk

Description

Transform example.com to example.co.uk

Priority

★ 100

Source

Any

Mode

AliasPatternMatch

Pattern type

Suffix

Pattern string

★ example.com

Pattern behavior

Replace

Replace string

example.co.uk

On successful match

Continue

Target zone

Head office

State

Enabled

Configuration

Rule name

★ Transform to example.net

Description

Transform example.com to example.net

Priority

★ 100

Source

Any

Mode

AliasPatternMatch

Pattern type

Suffix

Pattern string

★ example.com

Pattern behavior

Replace

Replace string

example.net

On successful match

Continue

Target zone

Head office

State

Enabled

About Call Policy

The VCS lets you set up rules to control which calls are allowed, which calls are rejected, and which calls are to be redirected to a different destination. These rules are known as Call Policy (or Administrator Policy).

If Call Policy is enabled and has been configured, each time a call is made the VCS will execute the policy in order to decide, based on the source and destination of the call, whether to

- proxy the call to its original destination
- redirect the call to a different destination or set of destinations
- reject the call.



When enabled, Call Policy is executed for all calls going through the VCS.

You can set up Call Policy in two ways:

- by [configuring basic Call Policy using the web interface](#) (note that this only lets you Allow or Reject specified calls)
- by [uploading a script written in the Call Processing Language](#) (CPL)



Only one of these two methods can be used at any one time to specify Call Policy. If a CPL script has been uploaded, this will disable use of the web interface to configure Call Policy. To use the web interface, you must delete the CPL script that has been uploaded.



Use **Call Policy** to determine which callers can make or receive calls via the VCS. Use [Allow and Deny lists](#) to determine which aliases can or cannot register with the VCS.

Call Policy and authentication

Call Policy uses the source and destination of a call to determine the action to be taken. Policy interacts with [authentication](#) when considering the source alias of the call. If your VCS is part of a secure environment, any policy decisions based on the source of the call should only be made when that source can be authenticated. Whether or not the VCS considers an endpoint to be authenticated depends on the **Authentication Mode** setting of the VCS.

Authentication mode off

When **Authentication Mode** is set to **Off**, calls will be accepted from any endpoint or neighbor. The assumption is that the source alias is trusted, so authentication is not required.

Authentication mode on

When [Authentication mode](#) is set to **On**, all endpoints and neighbors are required to authenticate with it before calls will be accepted. If a call is received from an unauthenticated source (e.g. neighbor or endpoint) the call's source aliases will be removed from the call request and replaced with an empty field before the Call Policy is executed. This is because there is a possibility that the source aliases could be forged and therefore they should not be used for policy decisions in a secure environment. This means that, when Authentication Mode is On and you configure policy based on the source alias, it will only apply to authenticated sources.

The VCS determines whether or not an endpoint is authenticated as follows:

H.323

When [Authentication mode](#) is set to **On**, for the purposes of Call Policy, an H.323 endpoint is considered to be **authenticated** if either of the following conditions apply:

- it is a locally registered endpoint. (Because **Authentication Mode** is **On**, the registration will have been accepted only after the endpoint authenticated successfully with the VCS.)
- it is a remote endpoint that is registered to and authenticated with a VCS that is a neighbor, traversal client or traversal server of the local VCS, and that remote VCS has in turn authenticated with the local VCS.

An H.323 endpoint is considered to be **unauthenticated** when:

- it is a remote endpoint registered to a neighbor and that neighbor has not authenticated with the VCS. This is regardless of whether or not the endpoint authenticated with the neighbor.

SIP

When [Authentication mode](#) is set to **On**, for the purposes of Call Policy a SIP endpoint is considered to be **authenticated** when:

- it falls within one of the domains for which the VCS is authoritative and has successfully responded to an authentication challenge. This endpoint could be registered to the local VCS or a VCS that is a traversal server or traversal client of the local VCS, as long as it is authoritative for the domain in the endpoint's AOR.

A SIP endpoint is considered to be **unauthenticated** if any of the following conditions apply:

- it does not fall within one of the domains for which the VCS is authoritative, or
- it has failed to successfully respond to an authentication challenge, or
- it has successfully responded to an authentication challenge but its **From** or **Reply-To** addresses are not compatible with the alias origin settings.

Enabling Call Policy

Call Policy is enabled and disabled using the **Call Policy mode** option on the **Call Policy configuration** page.

To go to the **Call Policy configuration** page:

- **VCS configuration > Call Policy > Configuration**

To enable Call Policy using the CLI:

- [xConfiguration Policy AdministratorPolicy Mode](#)

Call Policy mode options are:

On: Call Policy is enabled. If a CPL script has been uploaded, this policy will be used. Otherwise, the policy configured using the **Call Policy rules** page (**VCS configuration > Call Policy > Rules**) will be used.

Off: Call Policy is not in use.



You must click **Save** for any changes to the **Call Policy Mode** to take effect.



After you have enabled the use of Call Policy, you must define the policy to be used. This is done either [using the web interface](#) or by [uploading a CPL script](#).



If Call Policy is on but a policy has not been configured, then a default policy will be applied that allows all calls, regardless of source or destination.

Configuring basic Call Policy using the web interface

The **Call Policy rules** page lists the web-configured call policy rules currently in place, and lets you create, edit and delete simple Call Policy rules without having to write and upload a CPL script.

To go to the **Call Policy rules** page:

- **VCS configuration > Call Policy > Rules**



You cannot use the **Call Policy rules** page to configure Call Policy if a CPL file is already in place. If this is the case, on the **Call Policy configuration** page (**VCS configuration > Call Policy > Configuration**) you will have the option to **Delete uploaded file**. Doing so will delete the existing Call Policy that was put in place using a CPL script, and enable use of the **Call Policy rules** page for Call Policy configuration.

To add a new Call Policy rule, click **New**. You are taken to the **Add Call Policy rule** page.

To edit an existing rule, click **View/Edit**. You are taken to the **Edit Call Policy rule** page.

You can configure the following Call Policy rule options:

Source pattern

The alias that the calling endpoint used to identify itself when placing the call.

If this field is blank, the policy rule applies to all incoming calls from unauthenticated users, meaning calls where the endpoint making the call is **not** either:

- locally registered and authenticated with the VCS, or
- registered and authenticated to a neighbor which in turn has authenticated with the local VCS

See the [Call Policy and authentication](#) section for more information about unauthenticated users and Call Policy.

This field supports [regular expressions](#).

Destination pattern

The alias that the endpoint dialed to make the call.

This field supports [regular expressions](#).

Action

Whether or not a call that matches the source and destination is permitted.

Allow: if both the **Source** and **Destination** aliases match those listed, call processing will continue.

Reject: if both the **Source** and **Destination** aliases match those listed, the call will be rejected.

Rearrange

Each combination of **Source** and **Destination** is compared, in the order shown on the **Call Policy rules** page, with the details of the call being made until a match is found, at which point the call policy is applied.

To move a particular item to higher or lower in the list, thus giving the rule a higher or lower priority, click on the [↑](#) and [↓](#) icons respectively.

Configuring Call Policy using a CPL script

Overview

You can use CPL scripts to configure advanced Call Policy. To do this, you must first create and save the CPL script as a text file, after which you upload it to the VCS.



For information on the CPL syntax and commands that are supported by the VCS, see the [CPL reference](#) section.

Viewing existing CPL script

To view the Call Policy that is currently in place as an XML-based CPL script, go to the **Call Policy configuration** page (**VCS configuration > Call Policy > Configuration**) and click **Show Call Policy file**.

- If Call Policy has been configured using a CPL script, this shows you the script that was uploaded.
- If Call Policy has been configured using the **Call Policy rules** page, this shows you the CPL version of the call policy rules.
- If **Call Policy mode** is **On** but a policy has not been configured, this shows you a default CPL script that allows all calls.



You may want to view the file to take a backup copy of the Call Policy, or, if Call Policy has been configured using the **Call Policy rules** page you could take a copy of this CPL file to use as a starting point for a more advanced CPL script.



If Call Policy has been configured using the **Call Policy rules** page and you download the CPL file and then upload it back to the VCS without editing it, the VCS will recognize the file and automatically add each rule back into the **Call Policy rules** page.

About CPL XSD files

The CPL script must be in a format supported by the VCS. The **Call Policy** configuration page allows you to download the XML schemas which are used to check the script before it is uploaded to the VCS, so you can check in advance that your CPL script is valid. Two download options are available:

Show CPL XSD file

Displays in your browser the XML schema used for the CPL script.

Show CPL Extensions XSD file

Displays in your browser the XML schema used for additional CPL elements supported by the VCS.

Uploading a CPL script

To upload a new CPL file:

1. Go to the **Call Policy configuration** page (**VCS configuration > Call Policy > Configuration**).



The CPL script cannot be uploaded using the command line interface.

2. From the **Policy files** section, in the **Select the new Call Policy file** field, enter the file name or **Browse** to the CPL script you wish to upload.
3. Click **Upload File**.

Deleting an existing CPL script

If a CPL script has already been uploaded, a **Delete uploaded file** button will be visible. Click here to delete the file.

VCS and the Cisco AM GW

Overview

The Cisco TelePresence Advanced Media Gateway (Cisco AM GW) provides support for transcoding between standard codecs (such as H.264) and Microsoft RT Video to allow high definition calls between Microsoft Office Communicator (MOC) clients and Cisco endpoints.

Configuring the VCS to use the Cisco AM GW

A VCS Control can be configured to route calls to or from a Microsoft Office Communications Server (OCS) zone via the Cisco AM GW.

To use the Cisco AM GW you must first configure at least two zones:

- An OCS zone (a zone with a **Zone profile** set to **Microsoft Office Communications Server 2007**).
- A Cisco AM GW zone (a zone with a **Zone profile** set to **TANDBERG Advanced Media Gateway**). Note that a Cisco AM GW zone can be configured with up to six Cisco AM GW peers for load balancing purposes. Also, Cisco AM GW zones do not require any associated search rules.

To start using the Cisco AM GW to transcode calls:

1. Go to the **Advanced Media Gateway configuration** page (**VCS configuration > Advanced Media Gateway > Configuration**).
2. Choose the required Cisco AM GW zone from the **Advanced Media Gateway zone** drop-down. Note that only zones configured with a **Zone profile** of **TANDBERG Advanced Media Gateway** appear in this list. After a zone is selected, calls to or from the OCS are routed via the Cisco AM GWs connected to that zone.



By default, all OCS calls are routed via the Cisco AM GW but if you want to control which calls go through the Cisco AM GW you have to set up policy rules. To do this, set **Policy mode** to **On** and then follow the related task link to **Configure Advanced Media Gateway policy rules**.

To configure the required Cisco AM GW zone using the CLI:

- [xConfiguration Services AdvancedMediaGateway Zone Name](#)



For more information on configuring VCS and OCS, refer to the [Microsoft OCS 2007 \(R1 and R2\) and VCS Control Deployment Guide \[24\]](#).

Usage features and limitations

- If the Cisco AM GW reaches its capacity, any calls that would normally route via the Cisco AM GW will not fail; the call will still connect as usual but will not be transcoded.
- The OCS zone must be inside any firewall; the endpoint receiving or making the call can be outside the firewall.
- The VCS shows calls routed via the Cisco AM GW as two calls: one from the endpoint via the VCS to the Cisco AM GW which will be a local or traversal call as appropriate, and then a separate call back from the Cisco AM GW via the VCS to the OCS which will always be a local call.
- Bandwidth controls can be applied to the leg of the call between the endpoint and the Cisco AM GW zone, but cannot be applied to the Cisco AM GW zone to OCS zone leg of the call.

Controlling which calls to route via the Cisco AM GW

By default, when an **Advanced Media Gateway zone** has been selected, all OCS calls are routed via the Cisco AM GW.

If you want to control which calls go through the Cisco AM GW you have to set up policy rules. To do this:

1. Go to the **Advanced Media Gateway configuration** page (**VCS configuration > Advanced Media Gateway > Configuration**).
2. Set **Policy mode** to **On** and click **Save**.
3. You now need to [configure your policy rules](#).
Either follow the related task link to **Configure Advanced Media Gateway policy rules** or go to **VCS configuration > Advanced Media Gateway > Policy rules**.

To enable Cisco AM GW policy using the CLI:

- [xConfiguration Services AdvancedMediaGateway Policy Mode](#)

VCS and the Cisco AM GW (continued)

Configuring Cisco AM GW policy rules

The **Advanced Media Gateway policy rules** page (**VCS configuration > Advanced Media Gateway > Policy rules**) lists the set of rules that control which calls can go through the Cisco AM GW.

- By default, after a VCS Control has been configured with the Cisco AM GW to use for OCS calls, all calls to or from the OCS zone are routed via the Cisco AM GW.
- The rules on this page are only applied if the **Policy mode** on the **Advanced Media Gateway configuration** page is set to **On**.
- A rule is applied if it matches either the source or destination alias of a call. Note that if the aliases associated with a call do not match any of the policy rules, the call will be routed via the Cisco AM GW.

The page lists all the currently configured rules and lets you create, edit, delete, enable and disable rules. Note that:

- Priority 1 rules are applied first, followed by all priority 2, and so on.
- You can click on a column heading to sort the list, for example by **Rule name** or **Priority**.
- If you hover your mouse pointer over a rule, the rule description (if one has been defined) appears as a tooltip.

To configure Cisco AM GW policy rules using the CLI:

- [xConfiguration Services AdvancedMediaGateway Policy Rules](#)
- [xCommand AMGWPolicyRuleAdd](#)
- [xCommand AMGWPolicyRuleDelete](#)

Enabling and disabling policy rules

When you are making or testing configuration changes to your Cisco AM GW policy rules, you may want to temporarily enable or disable certain rules. You may also want to configure certain rules but only apply them occasionally.

You can do this by selecting the rule's check box and clicking **Enable** or **Disable** as appropriate. Any disabled rules still appear in the rules list but are ignored by the VCS when determining which calls are routed via the Cisco AM GW.

Configuration options

The configurable options are:

Rule name

The name assigned to the rule.

Description

An optional free-form description of the rule.

Priority

Sets the order in which the rules are applied. The rules with the highest priority (1, then 2, then 3 and so on) are applied first. Multiple rules with the same priority are applied in configuration order.

Pattern type

The way in which the **Pattern string** must match either the source or destination alias of the call. Options are:

Exact: the string must match the alias character for character.

Prefix: the string must appear at the beginning of the alias.

Suffix: the string must appear at the end of the alias.

Regex: treats the string as a regular expression.

Pattern string

Specifies the pattern against which the alias is compared.

Action

The action to take if the source or destination alias of the call matches this policy rule.

Allow: the call can connect via the Cisco AM GW.

Deny: the call can connect but it will not use Cisco AM GW resources.

State

Indicates if the rule is enabled or not.



Cisco AM GW policy rules support the use of Regular Expressions in the **Pattern String** field. See the [Regular expression reference](#) Appendix for more information.



You can test whether a pattern will match a particular alias and be transformed in the expected way by using the [Check pattern](#) tool (**Maintenance > Tools > Check pattern**).

Overview

A URI address typically takes the form *name@example.com*, where *name* is the alias and *example.com* is the domain.

URI dialing can make use of DNS to enable endpoints registered with different systems to locate and call each other. Without DNS, the endpoints would need to be registered to the same or neighbored systems in order to locate each other.

URI dialing without DNS

Without the use of DNS, calls made by a locally registered endpoint using URI dialing will be placed only if the destination endpoint is also locally registered, or is accessible via a neighbor system. This is because these endpoints would be located using the [zone search process](#), rather than a DNS query.

If you want to use URI dialing from your network without the use of DNS, you would need to ensure that all the systems in your network were connected to each other by neighbor relationships - either directly or indirectly. This would ensure that any one system could locate an endpoint registered to itself or any another system, by searching for the endpoint's URI.

This does not scale well as the number of systems grows. It is also not particularly practical, as it means that endpoints within your network will not be able to dial endpoints registered to systems outside your network (for example when placing calls to another company) if there is not already a neighbor relationship between the two systems.



If a DNS zone and a DNS server have not been configured on the local VCS, calls to endpoints that are not registered locally or to a neighbor system could still be placed if the local VCS is neighbored (either directly or indirectly) with another VCS that has been configured for URI dialing via DNS. In this case, any URI-dialed calls that are picked up by search rules that refer to that neighbor zone will go via that neighbor, which will perform the DNS lookup.

This configuration is useful if you want all URI dialing to be made via one particular system, such as a VCS Expressway.

If you do not want to use DNS as part of URI dialing within your network, then no special configuration is required. Endpoints will register with an alias in the form of a URI, and when calls are placed to that URI the VCS will query its local zone and neighbors for that URI.

If the VCS does not have DNS configured and your network includes H.323 endpoints, then in order for these endpoints to be reachable using URI dialing either:

- the H.323 endpoints should register with the VCS using an address in the format of a URI
- an appropriate transform should be written to convert URIs into the format used by the H.323 registrations. An example would be a deployment where H.323 endpoints register with an **alias**, and incoming calls are made to **alias@domain.com**. A local transform is then configured to strip the **@domain**, and the search is made locally for **alias**. See the [Stripping @domain for dialing to H.323 numbers](#) section for an example of how to do this.

SIP endpoints always register with an AOR in the form of a URI, so no special configuration is required.

URI dialing via DNS

By using DNS as part of URI dialing, it is possible to find an endpoint even though it may be registered to an unknown system. The VCS uses a DNS lookup to locate the domain in the URI address and then queries that domain for the alias.

The remainder of this section describes how to configure the VCS to support URI dialing via DNS.

Enabling URI dialing via DNS

URI dialing via DNS is enabled separately for outgoing and incoming calls.

Outgoing calls

To enable your VCS to locate endpoints using URI dialing via DNS, you must:

- configure at least one DNS zone and an associated search rule
- configure at least one DNS server

This is described in the [URI dialing via DNS for outgoing calls](#) section.

Incoming calls

To enable endpoints registered to your VCS to receive calls from non-locally registered endpoints using URI dialing via DNS, you must:

- ensure all endpoints are registered with an AOR (SIP) or H.323 ID in the form of a URI
- configure appropriate DNS records, depending on the protocols and transport types you wish to use.

This is described in the [URI dialing via DNS for incoming calls](#) section.

Firewall traversal calls

To configure your system so that you can place and receive calls using URI dialing through a firewall, see the [URI dialing and firewall traversal](#) section.

URI resolution process using DNS

When a VCS is attempting to locate a destination URI address using the DNS system, the general process is as follows:

H.323

1. The VCS will send a query to its DNS server(s) for an SRV record for the domain in the URI. (If more than one DNS server has been configured on the VCS, the query will be sent to all servers at the same time, and all responses will be prioritized by the VCS with only the most relevant SRV record being used.) If available, this SRV record will return information (e.g. the FQDN and listening port) about either the device itself or the authoritative H.323 gatekeeper for that domain.
 - If the domain part of the URI address was resolved successfully using an H.323 Location SRV record (i.e. for `_h3231s`) then the VCS will send an A/AAAA record query for each name record returned. These will resolve to one or more IP addresses, and the VCS then sends, in priority order, an LRQ for the full URI to those IP addresses.
 - If the domain part of the URI address was resolved using an H.323 Call Signaling SRV record (i.e. for `_h323cs`) then the VCS will send an A/AAAA record query for each name record returned. These will resolve to one or more IP addresses, and the then routes the call, in priority order to the IP addresses returned in those records. (An exception to this is where the original dial string has a port specified - e.g. `user@example.com:1719` - in which case the address returned is queried via an LRQ for the full URI address.).
2. If a relevant SRV record cannot be located:
 - If the **Include address record** setting for the DNS zone being queried is set to **On**, the system will fall back to looking for an A or AAAA record for the domain in the URI. If such a record is found, the call will be routed to that IP address and the search will terminate.
Note: if the A and AAAA records that are found at this domain are for systems other than those that support SIP or H.323, the VCS will still forward the call to this zone, and the call will therefore fail. For this reason, we recommend that this setting is left as the default **Off**.
 - If the **Include address record** setting for the DNS zone being queried is set to **Off**, the VCS will not query for A and AAAA records and instead will continue with the search, querying the remaining lower priority zones.

SIP

The VCS supports the SIP resolution process as outlined in [RFC 3263 \[16\]](#). An example of how the VCS implements this process is as follows:

1. The VCS will send a NAPTR query for the domain in the URI. If available, the result set of this query will describe a prioritized list of SRV records and transport protocols that should be used to contact that domain.
If no NAPTR records are present in DNS for this domain name then the VCS will use a default list of `_sips._tcp.<domain>`, `_sip._tcp.<domain>` and `_sip._udp.<domain>` for that domain as if they had been returned from DNS.
 - The VCS will send SRV queries for each result returned from the NAPTR record lookup. A prioritized list of A/AAAA records returned is built.
 - The VCS will send an A/AAAA record query for each name record returned by the SRV record lookup.

The above steps will result in a tree of IP addresses, port and transport protocols to be used to contact the target domain. The tree is sub-divided by NAPTR record priority and then by SRV record priority. When the tree of locations is used, the searching process will stop on the first location to return a response that indicates that the target destination has been contacted.
2. If the search process does not return a relevant SRV record:
 - If the **Include address record** setting for the DNS zone being queried is set to **On**, the system will fall back to looking for an A or AAAA record for the domain in the URI. If such a record is found, the call will be routed to that IP address and the search will terminate.
Note: if the A and AAAA records that are found at this domain are for systems other than those that support SIP or H.323, the VCS will still forward the call to this zone, and the call will therefore fail. For this reason, we recommend that this setting is left as the default **Off**.
 - If the **Include address record** setting for the DNS zone being queried is set to **Off**, the VCS will not query for A and AAAA records and instead will continue with the search, querying the remaining lower priority zones.

URI dialing via DNS for outgoing calls

URI dialing process

When a user places a call using URI dialing, they will typically dial an address in the form *name@example.com* from their endpoint. Below is the process that is followed when a URI address is dialed from an endpoint registered with your VCS, or received as a query from a neighbor system:

1. The VCS checks its [search rules](#) to see if any of them are configured with a **Mode** of either:
 - **Any Alias**, or
 - **Alias Pattern Match** with a pattern that matches the URI address
2. The associated target zones are queried, in rule priority order, for the URI.
 - If one of the target zones is a DNS zone, the VCS attempts to locate the endpoint through a DNS lookup. It does this by querying the DNS server configured on the VCS for the location of the domain as per the [URI resolution process via DNS](#). If the domain part of the URI address is resolved successfully the request is forwarded to those addresses.
 - If one of the target zones is a neighbor, traversal client or traversal server zones, those zones are queried for the URI. If that system supports URI dialing via DNS, it may route the call itself.

Adding and configuring DNS zones

To enable URI dialing via DNS, you must configure at least one DNS zone. To do this:

1. Go to the **Zones** page (**VCS configuration > Zones**).
2. Click **New**. You are taken to the **Create zone** page.
3. Enter a **Name** for the zone and select a **Type** of **DNS**.
4. Configure the DNS zone settings as described in the following sections.
5. Click **Create zone**.

To create a new DNS zone using the CLI:

- [xCommand ZoneAdd](#)

Hop count

When dialing by URI via DNS, the hop count used is that configured for the DNS zone associated with the search rule that matches the URI address (if this is lower than the hop count currently assigned to the call).

If URI address isn't matched to a DNS zone, the query may be forwarded to a neighbor. In this case, the hop count used will be that configured for the neighbor zone (if this is lower than the hop count currently assigned to the call).

H.323 and SIP modes

The H.323 and SIP sections allow you to filter calls to systems and endpoints located via this zone, based on whether the call is located using SIP or H.323 SRV lookups.

Include address record

This setting determines whether, if no NAPTR (SIP) or SRV (SIP and H.323) records have been found for the dialed alias via this zone, the VCS will then query for A and AAAA DNS records before moving on to query lower priority zones.

We recommend that this setting is left as the default **Off**, meaning that the VCS will not query for A and AAAA records, and instead will continue with the search, querying the remaining lower priority zones. This is because, unlike for NAPTR and SRV records, there is no guarantee that the A/AAAA records will point to a system capable of processing the relevant SIP or H.323 messages (LRQs, Setups, etc.) - the system may instead be a web server that processes http messages, or a mail server that processes mail messages. If this setting is **On**, when a system is found using A/AAAA lookup, the VCS will send the signaling to that destination and will not continue the search process. If the system does not support SIP or H.323, the call will fail.

Zone profile

For most deployments, this option should be left as **Default**.

Configuring search rules for DNS zones

If you want your local VCS to use DNS to locate endpoints outside your network, you should create a DNS zone and set up associated search rules that use the **Pattern string** and **Pattern type** fields to define the aliases that will trigger a DNS query.

For example, rules with:

- a **Pattern string** of `*.*@.*` and a **Pattern type** of **Regex** will query DNS for all aliases in the form of typical URI addresses
- a **Pattern string** of `(?!.*@example.com$).*` and a **Pattern type** of **Regex** will query DNS for all aliases in the form of typical URI addresses except those for the domain *example.com*

To set up further filters, configure extra search rules that target the same DNS zone. You do not need to create new DNS zones for each rule unless you want to filter based on the protocol (SIP or H.323) or use different hop counts.



You are not recommended to configure search rules with a **Mode** of **Any Alias** for DNS zones. This will result in DNS always being queried for all aliases, including those that may be locally registered and those that are not in the form of URI addresses.

Configuring DNS servers

To configure the DNS servers used by the VCS for DNS queries:

- **System configuration > DNS**. You are taken to the **DNS** page.
- [xConfiguration IP DNS Server](#)

Address 1 to Address 5

Enter the IP addresses of up to 5 DNS servers that the VCS will query when attempting to locate a domain. These fields must use an IP address, not a FQDN.



The DNS servers configured here are used as part of both the ENUM dialing and URI dialing via DNS processes.

If you do not configure any DNS servers, calls made using URI dialing will still be placed if the destination endpoint is locally registered or registered to a neighbor system, because locating these URIs does not require the use of DNS.

URI dialing via DNS for incoming calls

Types of DNS records required

The ability of the VCS to receive incoming calls made using URI dialing via DNS relies on the presence of DNS records for each domain the VCS is hosting.

These records can be of various types including:

- **A records**, which provide the IPv4 address of the VCS
- **AAAA records**, which provide the IPv6 address of the VCS
- **Service (SRV) records**, which specify the FQDN of the VCS and the port on it to be queried for a particular protocol and transport type.
- **NAPTR records**, which specify SRV record and transport preferences for a SIP domain.

You should provide an SRV or NAPTR record for each combination of domain hosted and protocol and transport type enabled on the VCS.

Incoming call process

When an incoming call has been placed using URI dialing via DNS, the VCS will have been located by the calling system using one of the DNS record lookups described above. The VCS will receive the request containing the dialed URI in the form *user@example.com*. This will appear as coming from the Default Zone. The VCS will then search for the URI in accordance with its normal [search process](#), applying any pre-search transforms, Call Policy and FindMe policy, then searching its Local Zone and other configured zones, in order of search rule priority.

SRV record format

The format of SRV records is defined by [RFC 2782 \[3\]](#) as:

_Service._Proto.Name TTL Class SRV Priority Weight Port Target

For the VCS, these will be as follows:

- **_Service** and **_Proto** will be different for H.323 and SIP, and will depend on the protocol and transport type being used.
- **Name** is the domain in the URI that the VCS is hosting (e.g. example.com)
- **Port** is the IP port on the VCS that has been configured to listen for that particular service and protocol combination
- **Target** is the FQDN of the VCS.

Configuring H.323 SRV records

Annex O of [H.323 \[15\]](#) defines the procedures for using DNS to locate gatekeepers and endpoints and for resolving H.323 URL aliases. It also defines parameters for use with the H.323 URL.

The VCS supports two types of SRV record as defined by this Annex. These are Location and Call, with **_Service** set to **_h323ls** and **_h323cs** respectively.

If you wish the VCS to be contactable using H.323 URI dialing, you should provide at least a Location SRV record, as it provides the most flexibility and the simplest configuration.

Location SRV records

For each domain hosted by the VCS, you should configure a Location SRV record as follows:

- **_Service** is **_h323ls**
- **_Proto** is **_udp**
- **Port** is the port number that has been configured from **VCS configuration > Protocols > H.323** as the **Registration UDP port**.

Call SRV records

Call SRV records (and A/AAAA records) are intended primarily for use by endpoints which cannot participate in a location transaction, exchanging LRQ and LCF. The configuration of a Call SRV record should be as follows:

- **_Service** is **_h323cs**
- **_Proto** is **_tcp**
- **Port** is the port number that has been configured from **VCS configuration > Protocols > H.323** as the **Call signaling TCP port**.

Configuring SIP SRV records

[RFC 3263 \[16\]](#) describes the DNS procedures used to resolve a SIP URI into the IP address, port, and transport protocol of the next hop to contact.

If you wish the VCS to be contactable using SIP URI dialing, you should configure an SRV record for each SIP transport protocol enabled on the VCS (i.e. UDP, TCP or TLS) as follows:

- Valid combinations of **_Service** and **_Proto** are:
 - **_sips._tcp**
 - **_sip._tcp**
 - **_sip._udp**
- **Port** is the IP port number that has been configured from **VCS configuration > Protocols > SIP** as the **port** for that particular transport protocol.

URI dialing via DNS for incoming calls

Example DNS record configuration

A company with the domain name **example.com** wants to enable incoming H.323 and SIP calls using URI addresses in the format **user@example.com**. The VCS hosting the domain has the FQDN **vcs.example.com**.

Their DNS records would typically be as follows:

- SRV record for **_h323ls._udp.example.com** returns **vcs.example.com**
- SRV record for **_h323cs._tcp.example.com** returns **vcs.example.com**
- NAPTR record for **example.com** returns
 - **_sip._tcp.example.com** and
 - **_sip._udp.example.com** and
 - **_sips._tcp.example.com**
- SRV record for **_sip._udp.example.com** returns **vcs.example.com**
- SRV record for **_sip._tcp.example.com** returns **vcs.example.com**
- SRV record for **_sips._tcp.example.com** returns **vcs.example.com**
- A record for **vcs.example.com** returns the IPv4 address of the VCS
- AAAA record for **vcs.example.com** returns the IPv6 address of the VCS

How you add the DNS records depends on the type of DNS server you are using. Instructions for setting up two common DNS servers are given in the [DNS configuration](#) Appendix.

In order for locally registered H.323 endpoints to be reached using URI dialing, either:

- the H.323 endpoints should register with the VCS using an address in the format of a URI
- an appropriate transform should be written to convert URIs into the format used by the H.323 registrations. An example would be a deployment where H.323 endpoints register with an *alias*, and incoming calls are made to *alias@domain.com*. A local transform is then configured to strip the *@domain*, and the search is made locally for *alias*. See the [Stripping @domain for dialing to H.323 numbers](#) section for an example of how to do this.

SIP endpoints always register with an AOR in the form of a URI, so no special configuration is required.



Several mechanisms could have been used to locate the VCS. You may wish to enable calls placed to *user@<IP_address>* to be routed to an existing registration for *user@example.com*. In this case you would configure a [pre-search transform](#) that would strip the *IP_address* suffix from the incoming URI and replace it with the suffix of *example.com*.

URI dialing and firewall traversal

Recommended configuration

If URI dialing via DNS is being used in conjunction with firewall traversal, DNS zones should be configured on the VCS Expressway and any VCSs on the public network only. VCSs behind the firewall should not have any DNS zones configured. This will ensure that any outgoing URI calls made by endpoints registered with the VCS will be routed through the VCS Expressway.

In addition, the DNS records for incoming calls should be configured with the address of the VCS Expressway as the authoritative gatekeeper/proxy for the enterprise (see the [DNS configuration](#) Appendix for more information). This ensures that incoming calls placed using URI dialing enter the enterprise through the VCS Expressway, allowing successful traversal of the firewall.

Overview

ENUM dialing allows an endpoint to be contacted by a caller dialing an E.164 number - a telephone number - even if that endpoint has registered using a different format of alias.

Using ENUM dialing, when an E.164 number is dialed it is converted into a URI using information stored in DNS. The VCS then attempts to find the endpoint based on the URI that has been returned.

The ENUM dialing facility allows you to retain the flexibility of URI dialing while having the simplicity of being called using just a number - particularly important if any of your callers are restricted to dialing using a numeric keypad.



The VCS supports outward ENUM dialing by allowing you to configure ENUM zones on the VCS. When an ENUM zone is queried, this triggers the VCS to transform the E.164 number that was dialed into an ENUM domain which is then queried for using DNS.

Note however that ENUM dialing relies on the presence of relevant DNS NAPTR records for the ENUM domain being queried. These are the responsibility of the administrator of that domain.

ENUM dialing process

When a VCS is attempting to locate a destination endpoint using ENUM, the general process is as follows:

1. The user dials the E.164 number from their endpoint.
2. The VCS converts the E.164 number into an ENUM domain as follows:
 - a. the digits are reversed and separated by a dot
 - b. the name of the domain that is hosting the NAPTR records for that E.164 number is added as a suffix.
3. DNS is then queried for the resulting ENUM domain.
4. If a NAPTR record exists for that ENUM domain, this will advise how the number should be converted into one (or possibly more) H.323/SIP URIs.
5. The VCS begins the search again, this time for the converted URI as per the [URI dialing process](#). Note that this is considered to be a completely new search, and so pre-search transforms and Call Policy will therefore apply.

Enabling ENUM dialing

ENUM dialing is enabled separately for incoming and outgoing calls.

Outgoing Calls

To allow locally registered endpoints to dial out to other endpoints using ENUM, you must

- configure at least one ENUM zone, and
- configure at least one DNS Server.

This is described in the [ENUM dialing for outgoing calls](#) section.

Incoming Calls

To enable endpoints in your enterprise to receive incoming calls from other endpoints via ENUM dialing, you must configure a DNS NAPTR record mapping your endpoints' E.164 numbers to their SIP/H.323 URIs. See the [ENUM dialing for incoming calls](#) section for instructions on how to do this.



If an ENUM zone and a DNS server have not been configured on the local VCS, calls made using ENUM dialing could still be placed if the local VCS is neighbored with another VCS that has been appropriately configured for ENUM dialing. Any ENUM dialed calls will go via the neighbor. This configuration is useful if you want all ENUM dialing from your enterprise to be configured on one particular system.

ENUM dialing for outgoing calls

Prerequisites

For a local endpoint to be able to dial another endpoint using ENUM via your VCS, the following conditions must be met:

- There must be a NAPTR record available in DNS that maps the called endpoint's E.164 number to its URI. It is the responsibility of the administrator of the enterprise to which the called endpoint belongs to provide this record, and they will only make it available if they wish the endpoints in their enterprise to be contactable via ENUM dialing.
- You must [configure an ENUM zone](#) on your local VCS. This ENUM zone must have a **DNS Suffix** that is the same as the domain where the NAPTR record for the called endpoint is held.
- You must [configure your local VCS with the address of at least one DNS server](#) that it can query for the NAPTR record (and if necessary any resulting URI).

After the ENUM process has returned one or more URIs, a new search will begin for each of these URIs in accordance with the [URI dialing process](#). If the URIs belong to locally registered endpoints, no further configuration is required. However, if one or more of the URIs are not locally registered, you may also need to [configure a DNS zone](#) if they are to be located using a DNS lookup.

Process

The process below is followed when an ENUM (E.164) number is dialed from an endpoint registered with your VCS:

1. The user dials the E.164 number from their endpoint.
2. The VCS initiates a search for the E.164 number as dialed. It follows the usual [search process](#).
3. After applying any pre-search transforms, the VCS checks its [search rules](#) to see if any of them are configured with a **Mode** of either:
 - **Any Alias**, or
 - **Alias Pattern Match** with a pattern that matches the E.164 number
4. The target zones associated with any matching search rules are queried in rule priority order.
 - If a target zone is a neighbor zone, the neighbor is queried for the E.164 number. If the neighbor supports ENUM dialing, it may route the call itself.
 - If a target zone is an ENUM zone, the VCS attempts to locate the endpoint through ENUM. As and when each ENUM zone configured on the VCS is queried, the E.164 number is transformed into an ENUM domain as follows:
 - a. The digits are reversed and separated by a dot
 - b. The **DNS Suffix** configured for that ENUM zone is appended.
5. DNS is then queried for the resulting ENUM domain.
6. If the DNS server finds at that ENUM domain a NAPTR record that matches the transformed E.164 number (i.e. after it has been reversed and separated by a dot), it returns the associated URI to the VCS.
7. The VCS then initiates a new search for that URI (maintaining the existing hop count). The VCS starts at the beginning of the search process (i.e. applying any pre-search transforms, then searching local and external zones in priority order). From this point, as it is now searching for a SIP/H.323 URI, the process for [URI dialing](#) is followed.

Example

In this example, we want to call Fred at Example Corp. Fred's endpoint is actually registered with the URI **fred@example.com**, but to make it easier to contact him his system administrator has configured a DNS NAPTR record mapping this alias to his E.164 number: **+44123456789**.

We know that the NAPTR record for example.com uses the DNS domain of **e164.arpa**.

1. We create an ENUM zone on our local VCS with a **DNS suffix** of **e164.arpa**.
2. We configure a search rule with a pattern match mode of **Any Alias**, and set the **Target zone** to the ENUM zone. This means that ENUM will always be queried regardless of the format of the alias being searched for.
3. We dial **44123456789** from our endpoint.
4. The VCS initiates a search for a registration of **44 118 123 456** and the search rule of **Any Alias** means the ENUM zone is queried. (Note that other higher priority searches could potentially match the number first.)
5. Because the zone being queried is an ENUM zone, the VCS is automatically triggered to transform the number into an ENUM domain as follows:
 - a. the digits are reversed and separated by a dot:
9.8.7.6.5.4.3.2.1.4.4
 - b. the **DNS Suffix** configured for this ENUM zone, **e164.arpa**, is appended.This results in a transformed domain of **9.8.7.6.5.4.3.2.1.4.4.e164.arpa**.
6. DNS is then queried for that ENUM domain.
7. The DNS server finds the domain and returns the information in the associated NAPTR record. This tells the VCS that the E.164 number we have dialed is mapped to the SIP URI of **fred@example.com**.
8. The VCS then starts another search, this time for **fred@example.com**. From this point the process for [URI dialing](#) is followed, and results in the call being forwarded to Fred's endpoint.

ENUM dialing for outgoing calls

Adding and configuring ENUM zones

For locally registered endpoints to use ENUM dialing, you must configure an ENUM zone for each ENUM service used by remote endpoints. To do this:

1. Go to the **Zones** page (**VCS configuration > Zones.**)
2. Click **New**. You are taken to the **Create zone** page.
3. Enter a **Name** for the zone and select a **Type** of **ENUM**.
4. Configure the ENUM zone settings as described in the following sections.
5. Click **Create zone**.

To create a new ENUM zone using the CLI:

- [xCommand ZoneAdd](#)
- [xConfiguration Zones Zone \[1..1000\]](#)



Any number of ENUM zones may be configured on the VCS. You should configure at least one ENUM zone for each DNS suffix that your endpoints may use.



Normal search rule pattern matching and prioritization rules apply to ENUM zones.

Hop count

When placing a call dialed using ENUM, the actual hop count used for the call is the hop count of the zone via which the call is eventually placed after the alias has been resolved.

DNS suffix

The DNS zone that is to be queried for a NAPTR record. This suffix is appended to the transformed E.164 number in an attempt to find a matching NAPTR record.

H.323 mode

Controls if H.323 records are looked up for this zone.

SIP mode

Controls if SIP records are looked up for this zone.

Configuring matches for ENUM zones

If you want locally registered endpoints to be able to make ENUM calls via the VCS, then at a minimum you should configure an ENUM zone and a related search rule with:

- a **DNS suffix** of **e164.arpa** (the domain specified by the ENUM standard)
- a related search rule with a **Mode** of **Any Alias**

This results in DNS always being queried for all types of aliases, not just ENUMs. It also means that ENUM dialing will only be successful if the enterprise being dialed uses the **e164.arpa** domain. To ensure successful ENUM dialing, you must configure an ENUM zone for each domain that holds NAPTR records for endpoints that callers in your enterprise might want to dial.

You can then set up search rules that filter the queries sent to each ENUM zone as follows:

- use a **Mode** of **Alias Pattern Match**
- use the **Pattern string** and **Pattern type** fields to define the aliases for each domain that will trigger an ENUM lookup

Example

For example, you want to enable ENUM dialing from your network to a remote office in the UK where the endpoints' E.164 numbers start with **44**. You would configure an ENUM zone on your VCS, and then an associated search rule with:

- **Mode** of **Alias Pattern Match**
- **Pattern string** of **44**
- **Pattern type** of **Prefix**

This will result in an ENUM query being sent to that zone only when someone dials a number starting with **44**.

Configuring transforms for ENUM zones

You can configure transforms for ENUM zones in the same way as any other zones (see the [Configuring search and zone transform rules](#) section for full information).

Any ENUM zone transforms are applied before the number is converted to an ENUM domain.

Example

For example, you want to enable ENUM dialing from your network to endpoints at a remote site using a prefix of **8** followed by the last 4 digits of the remote endpoints' E.164 number. You would configure an ENUM zone on your VCS and then an associated search rule with:

- **Mode** of **Alias Pattern Match**
- **Pattern string** of **8(\d{4})**
- **Pattern type** of **Regex**
- **Pattern behavior** of **Replace**
- **Replace string** of **44123123(\1)**

With this configuration, it will be the resulting string (i.e. **44123123xxxx**) that is converted into an ENUM domain and queried for via DNS.



To verify you have configured your outward ENUM dialing correctly, use the **Locate** tool (**Maintenance > Tools > Locate**) to try to resolve an E.164 alias.

This can also be done using the CLI using [xCommand Locate](#).

Configuring DNS servers

To configure the VCS with details of DNS servers to be used when searching for NAPTR records:

- **System configuration > DNS.**
You are taken to the **DNS** page.
- [xConfiguration IP DNS Server](#)



For endpoints registered to the VCS to make outgoing calls using ENUM dialing, you must configure at least one DNS server for the VCS to query. For resilience, you can specify up to five DNS servers.



The DNS servers configured using this page are used as part of both the ENUM dialing and URI dialing via DNS processes.

Address 1 to Address 5

Enter the IP addresses of up to 5 DNS servers that the VCS will query when attempting to locate a domain.

ENUM dialing for incoming calls

Prerequisites

In order for your locally registered endpoints to be reached using ENUM dialing, you must configure a DNS NAPTR record that maps your endpoints' E.164 numbers to their SIP/H.323 URIs. This record must be located at an appropriate DNS domain where it can be found by any systems attempting to reach you by using ENUM dialing.

About DNS domains for ENUM

ENUM relies on the presence of NAPTR records to provide the mapping between E.164 numbers and their SIP/H.323 URIs. [RFC 3761 \[8\]](#), which is part of a suite of documents that define the ENUM standard, specifies that the domain for ENUM - where the NAPTR records should be located for public ENUM deployments - is **e164.arpa**. However, use of this domain requires that your E.164 numbers are assigned by an appropriate national regulatory body. Not all countries are yet participating in ENUM, so you may wish to use an alternative domain for your NAPTR records. This domain could reside within your corporate network (for internal use of ENUM) or it could use a public ENUM database such as <http://www.e164.org>.

Configuring DNS NAPTR records

ENUM relies on the presence of NAPTR records, as defined by [RFC 2915 \[7\]](#). These are used to obtain an H.323 or SIP URI from an E.164 number.

The record format that the VCS supports is:

- **order flag preference service regex replacement**
- where:

- **order** and **preference** determine the order in which NAPTR records are processed. The record with the lowest **order** is processed first, with those with the lowest **preference** being processed first in the case of matching **order**.
- **flag** determines the interpretation of the other fields in this record. Only the value **u** (indicating that this is a terminal rule) is currently supported, and this is mandatory.
- **service** states whether this record is intended to describe E.164 to URI conversion for H.323 or for SIP. Its value must be either **E2U+h323** or **E2U+SIP**.
- **regex** is a regular expression that describes the conversion from the given E.164 number to an H.323 or SIP URI.
- **replacement** is not currently used by the VCS and should be set to **.** (i.e. the full stop character).



Non-terminal rules in ENUM are not currently supported by the VCS. For more information on these, see section 2.4.1 of [RFC 3761 \[8\]](#).

Example

For example, the record:

- **IN NAPTR 10 100 "u" "E2U+h323" "!^(.*)\$!h323:\1@example.com!" .**

would be interpreted as follows:

- 10 is the **order**
- 100 is the **preference**
- **u** is the **flag**
- **E2U+h323** states that this record is for an H.323 URI
- **!^(.*)\$!h323:\1@example.com!** describes the conversion:
 - **!** is a field separator
 - the first field represents the string to be converted. In this example, **^(.*)\$** represents the entire E.164 number
 - the second field represents the H.323 URI that will be generated. In this example, **h323:\1@example.com** states that the E.164 number will be concatenated with **@example.com**. For example, **1234** will be mapped to **1234@example.com**.
- **.** shows that the **replacement** field has not been used.

Call routed mode

Overview

Calls are made up of two components - signaling and media.

For traversal calls, the VCS will always handle both the media and the signaling.

For non-traversal calls, the VCS will not handle the media, and may or may not need to handle the signaling. You can nominate whether or not the VCS will handle the signaling when it is not required to from the **Call routed mode** setting.



For a definition of a traversal call, see the [What are traversal calls?](#) section.

To configure the **Call routed mode** from the web interface, go to the **Calls** page (**VCS configuration > Calls**).

To configure this setting from the CLI:

- [xConfiguration Call Routed Mode](#)

The options for this setting are:

Always

The VCS will always handle the call signaling. The call will consume either a traversal call license (if it is a traversal call) or a local (non-traversal) call license (if it is not a traversal call) on the VCS.

Optimal

The VCS will handle the call signaling when the call is one of:

- a traversal call
- an H.323 call that has been modified by Call Policy or FindMe such that it resolves to more than one alias, or has a "no answer" or "busy" device configured
- one of the endpoints in the call is locally registered.

In all other cases the VCS will remove itself from the call signaling path after the call has been set up. The VCS will not consume a call license for any such calls, and the call signaling path will be simplified. This setting is useful in a [hierarchical dial plan](#), when used on the directory VCS. In such deployments the directory VCS is used to look up and locate endpoints and it does not have any endpoints registered directly to it.

Call loop detection mode

Overview

Your dial plan or that of networks to which you are neighbored may be configured in such a way that there are potential signaling loops. An example of this is a [structured dial plan](#), where all systems are neighbored together in a mesh. In such a configuration, if the [hop counts](#) are set too high, a single search request may be sent repeatedly around the network until the hop count reaches 0, consuming resources unnecessarily.

The VCS can be configured to detect search loops within your network and terminate such searches. This is done using the **Call loop detection mode** setting.

To configure the **Call loop detection mode** using the web interface, go to the **Calls** page (**VCS configuration > Calls**).

To configure this setting using the CLI:

- [xConfiguration Call Loop Detection Mode](#)

The options for this setting are:

On

The VCS will fail any branch of a search that contains a loop, recording it as a level 2 "loop detected" event.

Two searches will be considered to be a loop if they:

- have same call tag
- are for the same destination alias
- use the same protocol, and
- originate from the same zone.

Using this setting will allow you to save on network resources and fail call branches early where loops have been detected.

Off

The VCS will not detect and fail search loops. We recommend that you use this setting only in advanced deployments.



The loop detection feature was introduced in VCS version X4. It is only supported in deployments where all VCSs are running on X4 software or later.

Identifying calls

Each call that passes through the VCS is assigned a Call ID and a Call Serial Number. Calls also have a Call Tag assigned if this does not already exist.

Call ID

The VCS assigns each call currently in progress a different Call ID. The Call ID numbers start at 1 and go up to the maximum number of calls allowed on that system.

Each time a call is made, the VCS will assign that call the lowest available Call ID number. For example, if there is already a call in progress with a Call ID of 1, the next call will be assigned a Call ID of 2. If Call 1 is then disconnected, the third call to be made will be assigned a Call ID of 1.

The Call ID is not therefore a unique identifier: while no two calls in progress at the same time will have the same Call ID, the same Call ID will be assigned to more than one call over time.

Call Serial Number

The VCS assigns a unique Call Serial Number to every call passing through it. No two calls on a VCS will ever have the same Call Serial Number. A single call passing between two or more VCSs will be identified by a different Call Serial Number on each system.

Call Tag

Call Tags are used to track calls passing through a number of VCSs. When the VCS receives a call, it checks to see if there is a Call Tag already assigned to it. If so, the VCS will use the existing Call Tag; if not, it will assign a new Call Tag to the call. This Call Tag is then included in the call's details when the call is forwarded on. A single call passing between two or more VCSs will be assigned a different Call Serial Number each time it arrives at a VCS (including one it has already passed through) but can be identified as the same call by use of the Call Tag. This is particularly useful if you are using a [remote syslog server](#) to collate events across a number of VCSs in your network.

The Call Tag also helps identify loops in your network - it is used as part of the automatic [Call loop detection](#) feature, and you can also search the Event Log for all events relating to a single call tag. Loops occur when a query is sent to a neighbor zone and passes through one or more systems before being routed back to the original VCS. In this situation the outgoing and incoming query will have different Call Serial Numbers and may even be for different destination aliases (depending on whether any transforms were applied). However, the call will still have the same Call Tag.



Call Tags are supported by VCS version X3.0 or later. If a call passes through a system that is not a VCS, or a VCS that is running an earlier version of the software, the Call Tag information will be lost.

Identifying calls in the web interface

The **Call summary**, **Call details** and **Search details** pages shows the Call Tag. The **Call details** and **Search details** pages also show the Call Serial Number.



The VCS web interface does not show the Call ID.

Identifying calls in the CLI

To control a call using the CLI, you must reference the call using either its Call ID or Call Serial Number. These can be obtained using the command:

• [xStatus Calls](#)

This will return details of each call currently in progress in order of their Call ID. The second line of each entry will list the Call Serial Number, and the third will list the Call Tag.

```
*s Calls:
Call 5:
  SerialNumber: "7055fe80-225d-11b2-9527-0010f30f52"
  Tag: "7055ff70-225d-11b2-8f85-0010f30f5250"
  State: Connected
  StartTime: "2008-06-03 17:10:49"
  Duration: 11
  Legs:
    Leg 1:
      Protocol: H323
      H323:
        CallSignalAddress: "16.44-1-1:11017"
        Aliases:
          Alias 1:
            Type: H323Id
            Value: ""
```

Call ID

Call serial number

Call Tag

Disconnecting a call using the web interface

To disconnect one or more existing calls using the web interface:

1. Go to **Status > Calls**.
You will be taken to the **Calls** page.
2. If you want to confirm the details of the call, including the Call Serial Number and Call Tag, click **View**. Click the back button on your browser to return to the **Calls** page.
3. Select the box next to the call(s) you want to terminate and click **Disconnect**.



When disconnecting a call, only the call with that Call Serial Number will be disconnected. Other calls with the same Call Tag but different Call Serial Number may not be affected.

Disconnecting a call using the CLI

To disconnect an existing call using the CLI, you must first [obtain either the call ID number or the call serial number](#). Then use either one of the following commands as appropriate:

- `xCommand DisconnectCall Call: <ID number>`
- `xCommand DisconnectCall CallSerialNumber: <serial number>`

While it is quicker to use the call ID number to reference the call to be disconnected, there is a risk that in the meantime the call has already been disconnected and the call ID assigned to a new call. For this reason, the VCS also allows you to reference the call using the longer but unique call serial number.



When disconnecting a call, only the call with that Call Serial Number will be disconnected. Other calls with the same Call Tag but different Call Serial Number may not be affected.

Limitations when disconnecting SIP calls

The call disconnection API works differently for H.323 and SIP calls due to differences in the way the protocols work.

For H.323 calls, and interworked calls, the **Disconnect** command will actually disconnect the call.

For SIP calls, the **Disconnect** command will cause the VCS to release all resources used for the call and the call will appear on the system as disconnected. However, SIP calls are peer-to-peer and as a SIP proxy the VCS has no authority over the endpoints. Although releasing the resources may have the side-effect of disconnecting the SIP call, it is also possible that the call signaling, media or both may stay up (depending on the type of call being made). The call will not actually disconnect until the SIP endpoints involved have also cleared their resources.



Endpoints that support SIP session timers ([RFC 4028 \[14\]](#)) have a call refresh timer which allows them to detect a hung call (signaling lost between endpoints).

The endpoints can then release their resources after the negotiated timeout period.

Bandwidth control

This section describes the pages that appear under the **Local Zone** and **Bandwidth** sub-menus of the **VCS Configuration** menu in the web interface.

These pages allow you to control the bandwidth that is used for calls within your local zone, as well as calls out to other zones.



Bandwidth control on the VCS

The VCS allows you to control the amount of bandwidth used by endpoints on your network. This is done by grouping endpoints into subzones, and then applying limits to the bandwidth that can be used:

- within each subzone
- between a subzone and another subzone
- between a subzone and a zone

Bandwidth limits may be set on a call-by-call basis and/or on a total concurrent usage basis. This flexibility allows you to set appropriate bandwidth controls on individual components of your network.

This section describes the different types of subzones and how to add and configure them, and explains how to use [Links](#) and [Pipes](#) to apply bandwidth controls between subzones and zones.



Calls will fail if links are not configured correctly. You can check whether a call will succeed, and what bandwidth will be allocated to it, using the command [xCommand CheckBandwidth](#).



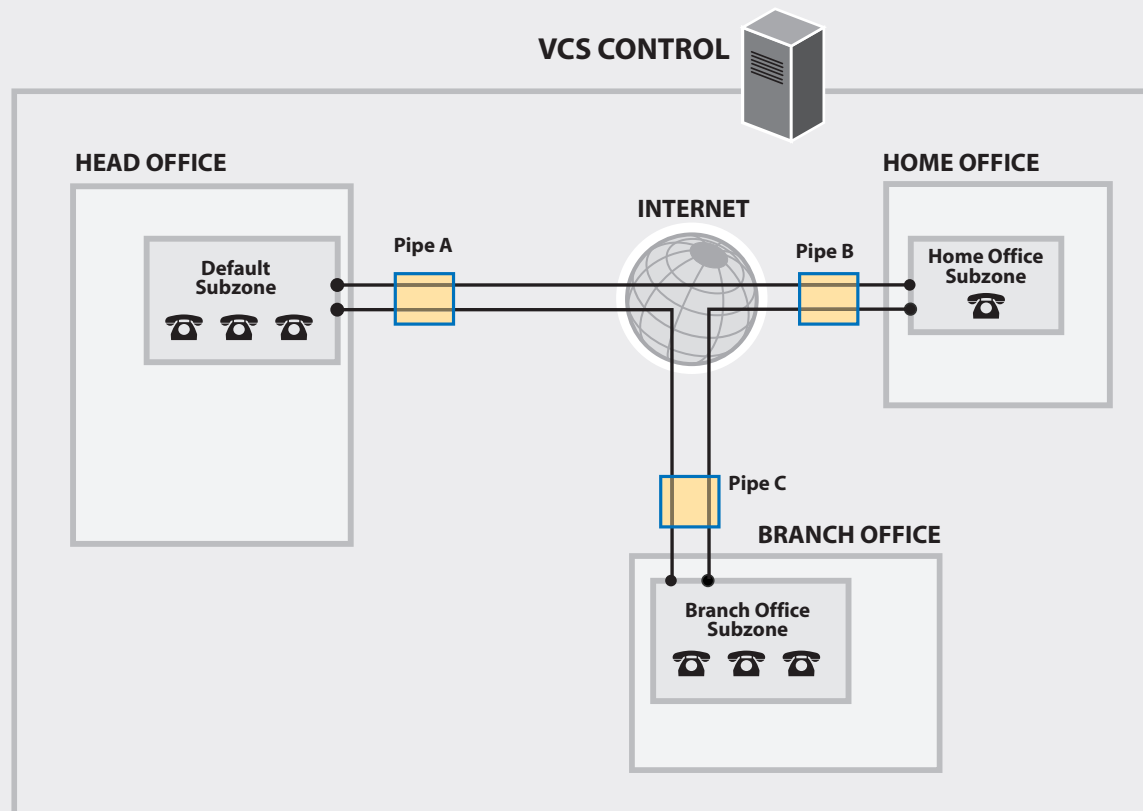
For specific information about how bandwidth is managed across peers in a cluster, see the [Sharing bandwidth across peers](#) section.

Example network deployment

The diagram below shows a typical network deployment:

- a broadband LAN between the Enterprise and the internet, where high bandwidth calls are acceptable
- a pipe to the internet (Pipe A) with restricted bandwidth
- two satellite offices, Branch and Home, each with their own internet connections and restricted pipes

In this example we have created new subzones for each pool of endpoints, so that we can apply suitable limitations to the bandwidth used within and between each subzone based on the amount of bandwidth they have available via their internet connections.



About subzones

The Local Zone is made up of subzones. Subzones are used to control the bandwidth used by various parts of your network, and to control registrations.

Three special subzones — the Default Subzone, the Traversal Subzone and the Cluster Subzone (only applies if the VCS is in a cluster) — are automatically created and cannot be deleted.

Additional subzones can also be manually configured.

When an endpoint registers with the VCS it is allocated to an appropriate subzone, determined by subzone membership rules based on endpoint IP address ranges or alias pattern matches.

See the [Configuring subzones and membership rules](#) section for more information.

Subzone registration policy

In addition to using [Allow and Deny Lists](#) to control whether an endpoint can register with the VCS, you can also configure each manually created subzone and the Default Subzone as to whether it will accept registrations assigned to it via the subzone membership rules.

This provides additional flexibility when defining your registration policy. For example you can:

- Deny registrations based on IP address subnet. You can do this by creating a subzone with associated membership rules based on an IP address subnet range, and then setting that subzone to deny registrations.
- Configure the Default Subzone to deny registrations. This would cause any registration requests that do not match any of the subzone membership rules, and hence fall into the Default Subzone, to be denied.

Note that registration requests have to fulfill any Allow/Deny List policy rules before any subzone membership and subzone registration policy rules are applied.

About the Traversal Subzone

The Traversal Subzone is a conceptual subzone. No endpoints can be registered to the Traversal Subzone; its sole purpose is to control the bandwidth used by traversal calls.

All traversal calls are deemed to pass through the Traversal Subzone, so by applying bandwidth limitations to the Traversal Subzone you can control how much processing of media the VCS will perform at any one time. These limitations can be applied on a total concurrent usage basis, and on a per-call basis.

Traversal calls

A traversal call is a call passing through the VCS that includes both the signaling (information about the call) and media (voice and video). The only other type of call is a non-traversal call, where the signaling passes through the VCS but the media goes directly between the endpoints (or between an endpoint and another VCS, or between two VCSs, in the call route).

The following types of calls require the VCS to take the media:

- firewall traversal calls, where the local VCS is either the traversal client or traversal server
- calls that are gatewayed (interworked) between H.323 and SIP on the local VCS
- calls that are gatewayed (interworked) between IPv4 and IPv6 on the local VCS
- for VCSs with Dual Network Interfaces enabled, calls that are inbound from one LAN port and outbound on the other
- a SIP to SIP call when one of the participants is behind a NAT (unless both endpoints are using [ICE](#) for NAT traversal)

All such calls require a traversal call license each time they pass through the Traversal Subzone.



A call is “traversal” or “non-traversal” from the point of view of the VCS through which it is being routed at the time. A call between two endpoints may pass through a series of VCSs. Some of these systems may just take the signaling, in which case the call will be a non-traversal call for that VCS. Other systems in the route may need to take the media as well, and so the call will count as a traversal call on that particular VCS.

About the Default Subzone

When an endpoint registers with the VCS, its IP address and alias is checked against the subzone membership rules and it is assigned to the appropriate subzone. If no subzones have been created, or the endpoint's IP address or alias do not match any of the subzone membership rules, it is assigned to the Default Subzone (providing the Default Subzone's **Registration policy** is set to **Allow**).

The use of a Default Subzone on its own (without any other manually configured subzones) is suitable only if you have uniform bandwidth available between all your endpoints. However, it is possible for a Local Zone to contain two or more different networks with different bandwidth limitations. In this situation, you should configure separate subzones for each different part of the network.

Use the **Default Subzone** page (**VCS configuration > Local Zone > Default Subzone**) to place bandwidth restrictions on calls involving endpoints in the Default Subzone, and to specify the Default Subzone's registration policy.

Default links between subzones

The VCS is shipped with the Default Subzone and Traversal Subzone (and [Default Zone](#)) already created, and with links between the three. You can delete or amend these default links if you need to model restrictions of your network.

If any of these links have been deleted, they can be automatically restored by using:

- [xCommand DefaultLinksAdd](#)

To restore these links from the web interface, you must recreate them manually. See the [Creating and editing links](#) section for instructions on how to do this.

Configuring subzones and membership rules

Configuring subzones

The **Subzones** page lists all the existing manually configurable subzones and lets you add, edit or delete subzones.

For each subzone, it shows how many membership rules it has, how many devices are currently registered to it, and the current number of calls and bandwidth in use.

To go to the **Subzones** page:

- **VCS configuration > Local Zone > Subzones.**

Click on the subzone you want to configure (or click **New** to create a new subzone, or click **Delete** to remove a subzone).

Up to 1000 subzones can be configured.

The configurable options are:

Name

Enter the name you want to assign to the subzone. This name is used when creating links.

Registration policy

Controls whether registrations assigned to the subzone are accepted. See [Subzone registration policy](#) for more details.

Bandwidth

See the [Applying bandwidth limitations to subzones](#) section for a description of these fields.

To add a new subzone using the CLI:

- **xCommand SubZoneAdd**

To configure an existing subzone using the CLI:

- **xConfiguration Zones LocalZone SubZones**

Configuring subzone membership rules

The **Subzone membership rules** page lists all the existing subzone membership rules and lets you add, edit, delete, enable or disable rules.

Each rule specifies the criteria used to match an endpoint registering with the VCS into an appropriate subzone.

To go to the **Subzone membership rules** page:

- **VCS configuration > Local Zone > Subzone membership rules.**



You can click on a column heading to sort the list, for example by **Priority** or **Subzone**. If you hover your mouse pointer over a rule, the rule description (if one has been defined) appears as a tooltip.

Click on the rule you want to configure (or click **New** to create a new rule, or click **Delete** to remove a rule).

Up to 3000 subzone membership rules can be configured.

Enabling and disabling membership rules

When you are making or testing configuration changes to your membership rules, you may want to temporarily enable or disable certain rules. You can do this by selecting the rule's check box and clicking **Enable** or **Disable** as appropriate. Any disabled rules still appear in the rules list but are ignored by the VCS when processing registration requests.

The configurable options are:

Rule name

A descriptive name for the membership rule.

Description

An optional free-form description of the rule.

Priority

The order in which the rules are applied (and thus to which subzone the endpoint is assigned) if an endpoint's address satisfies multiple rules.

The rules with the highest priority (1, then 2, then 3 and so on) are applied first. If multiple **Subnet** rules have the same priority the rule with the largest prefix length is applied first.

Alias Pattern Match rules at the same priority are searched in configuration order.

Type

The type of address that applies to this rule:

Subnet: assigns the device if its IP address falls within the configured IP address subnet.

Alias Pattern Match: assigns the device if its alias matches the configured pattern.



Pattern matching is useful, for example, for home workers on dynamic IP addresses; rather than having to continually update the subnet to match what has been allocated, you can match against their alias instead.

Subnet address and prefix length

These two fields together determine the range of IP addresses that will belong to this subzone. (Applies only if the **Type** is **Subnet**.)

The **Address range** field shows the range of IP addresses to be allocated to this subzone, based on the combination of the **Subnet address** and **Prefix length**.

Pattern type

How the pattern string must match the alias for the rule to be applied. (Applies only if the **Type** is **Alias Pattern Match**.)

Exact: the entire string must exactly match the alias character for character.

Prefix: the string must appear at the beginning of the alias.

Suffix: the string must appear at the end of the alias.

Regex: the string is treated as a regular expression.

Pattern string

The pattern against which the alias is compared. (Applies only if the **Type** is **Alias Pattern Match**.)

Target subzone

The subzone to which an endpoint is assigned if its address satisfies this rule.

State

Indicates if the rule is enabled or not.

To add a new membership rule using the CLI:

- **xConfiguration SubZoneMembershipRuleAdd**

To configure membership rules using the CLI:

- **xConfiguration Zones LocalZone SubZones MembershipRules**



If an endpoint's IP address or alias does not match any of the membership rules, it is assigned to the Default Subzone.

Applying bandwidth limitations to subzones

Types of limitations

You can apply bandwidth limits to the Default Subzone, Traversal Subzone and all manually configured subzones. The types of limitations you can apply vary depending on the type of subzone, as follows:

Limitation	Description	Can be applied to
Total	Limits the total concurrent bandwidth being used by all endpoints in the subzone at any one time. In the case of the Traversal Subzone, this is the maximum bandwidth available for all concurrent traversal calls.	- Default Subzone - Traversal Subzone - Manually configured subzones
Calls entirely within...	Limits the bandwidth of any individual call between two endpoints within the subzone.	- Default Subzone - Manually configured subzones
Calls into or out of...	Limits the bandwidth of any individual call between an endpoint in the subzone, and an endpoint in another subzone or zone.	- Default Subzone - Manually configured subzones
Calls handled by...	The maximum bandwidth available to any individual traversal call.	Traversal Subzone

For all these settings, a **bandwidth mode** of:

- **NoBandwidth** means that no bandwidth is allocated and therefore no calls can be made.
- **Limited** means that limits are applied. You must also enter a value in the corresponding **bandwidth (kbps)** field.
- **Unlimited** means that no restrictions are applied to the amount of bandwidth being used.



Use subzone bandwidth limits if you want to configure the bandwidth available between one specific subzone and **all other** subzones or zones.

Use **pipes** if you want to configure the bandwidth available between one specific subzone and **another specific** subzone or zone.

If your bandwidth configuration is such that multiple types of bandwidth restrictions are placed on a call (for example, if there are both subzone bandwidth limits and pipe limits), the lowest limit will always apply to that call.

How different bandwidth limitations are managed

In situations where there are differing bandwidth limitations applied to the same link, the lower limit will always be the one used when routing the call and taking bandwidth limitations into account.

For example, Subzone A may have a per call inter bandwidth of 128. This means that any calls between Subzone A and any other subzone or zone will be limited to 128kbps. However, Subzone A also has a link configured between it and Subzone B. This link uses a pipe with a limit of 512kbps. In this situation, the lower limit of 128kbps will apply to calls between the two, regardless of the larger capacity of the pipe.

In the reverse situation, where Subzone A has a per call inter bandwidth limit of 512kbps and a link to Subzone B with a pipe of 128, any calls between the two subzones will still be limited to 128kbps.

Bandwidth consumption of traversal calls

A **non-traversal call** between two endpoints within the same subzone would consume from that subzone the amount of bandwidth of that call. A **traversal call** between two endpoints within the same subzone must, like all traversal calls, pass through the Traversal Subzone. This means that such calls consume an amount of bandwidth from the originating subzone's total concurrent allocation that is equal to twice the bandwidth of the call – once for the call from the subzone to the Traversal Subzone, and again for the call from the Traversal Subzone back to the originating subzone.

In addition, as this call passes through the Traversal Subzone, it will consume an amount of bandwidth from the Traversal Subzone equal to that of the call.

About links

Subzones are connected to other subzones and zones via links. For a call to take place, the endpoints involved must each reside in subzones or zones that have a link between them. The link does not need to be direct; the two endpoints may be linked via one or more intermediary subzones.

Links are used to calculate how a call is routed over the network and therefore which zones and subzones are involved and how much bandwidth is available. If multiple routes are possible, your VCS will perform the bandwidth calculations using the one with the fewest links.

You can configure up to 3000 links.

Default links

If a subzone has no links configured, then endpoints within the subzone will only be able to call other endpoints within the same subzone. For this reason, when a subzone is created, it is automatically given certain links. See the [Automatically created links](#) section for more information.

Creating and editing links

Creating a new link

To create a new link:

- **VCS configuration > Bandwidth > Links.**
You will be taken to the **Links** page.
Click **New**.
You will be taken to the **Create link** page.

- [xCommand LinkAdd](#)

Editing an existing link

To edit a link:

- **VCS configuration > Bandwidth > Links.**
You will be taken to the **Links** page.
Click **View/Edit**.
You will be taken to the **Edit link** page.

- [xConfiguration Bandwidth Link](#)

The configurable options are:

Name

The name you want to assign to this link.

Node 1, Node 2

Select the names of the two subzones, or the subzone and zone, between which you want to create a link.

Pipe 1, Pipe 2

If you want to apply bandwidth limitations to this link, select the pipe(s) to be applied.

For more information, see the [Applying pipes to links](#) section.

Default links

About default links

If a subzone has no links configured, then endpoints within the subzone will only be able to call other endpoints within the same subzone. For this reason, the VCS comes shipped with a set of pre-configured links and will also automatically create new links each time you create a new subzone.



Calls will fail if links are not configured correctly. You can check whether a call will succeed, and what bandwidth will be allocated to it, using the command [xCommand CheckBandwidth](#).

Pre-configured links

The VCS is shipped with the Default Subzone, Traversal Subzone and Default Zone already created, and with default links pre-configured between the three which are named as follows:

- **DefaultSZtoTraversalSZ**
- **DefaultSZtoDefaultZ**
- **TraversalSZtoDefaultZ**



You can edit any of these default links in the same way you would edit manually configured links. See [Creating and editing links](#) for more information.

If any of these links have been deleted, they may all be automatically restored using:

- [xCommand DefaultLinksAdd](#)

To restore these links using the web interface, you must re-create them manually. See the [Creating and editing links](#) section for instructions on how to do this.

Automatically created links

Whenever a new subzone or zone is created, links are automatically created as follows:

New zone/subzone type	Default links are created to...
Subzone	Default Subzone and Traversal Subzone
Neighbor zone	Default Subzone and Traversal Subzone
DNS zone	Default Subzone and Traversal Subzone
ENUM zone	Default Subzone and Traversal Subzone
Traversal client zone	Traversal Subzone
Traversal server zone	Traversal Subzone

Along with the pre-configured default links this ensures that, by default, any new subzone or zone has connectivity to all other subzones and zones. You may rename, delete and amend any of these default links. See the [Creating and editing links](#) section for more information.

About pipes

It is possible to control the amount of bandwidth used on calls between specific subzones and zones. The limits can be applied to the total concurrent bandwidth used at any one time, or to the bandwidth used by any individual call.

To apply these limits, you create a pipe and configure it with the required bandwidth limitations. Then when configuring links you assign the pipe to one or more links. Calls using the link will then have the pipe's bandwidth limitations applied to them.

You can configure up to 1000 pipes.

See the [Applying pipes to links](#) section for more information.

Creating and editing pipes

Creating a new pipe

To create a pipe:

- **VCS configuration > Bandwidth > Pipes.**
You will be taken to the **Pipes** page.
Select **New**.
You will be taken to the **Create pipe** page.
- [xCommand PipeAdd](#)

Editing an existing pipe

To configure details of a pipe:

- **VCS configuration > Bandwidth > Pipes**
You will be taken to the **Pipes** page.
Click on the pipe you wish to configure.
You will be taken to the **Edit pipe** page.
- [xConfiguration Bandwidth Pipe](#)

Pipe configuration options

Name

Enter the name you want to give to this pipe. You will refer to this name when creating links.

Bandwidth restriction

Determines whether there is a limit on the total concurrent bandwidth of this pipe.

Unlimited: no limitations are in place.

Limited: there is a limit in place; you must enter the limit in the field below.

NoBandwidth: there is no bandwidth available.

Total bandwidth limit (kbps)

Sets the limit on the total concurrent bandwidth of this pipe.

Bandwidth restriction

Determines whether there is a limit on the bandwidth of individual calls via this pipe.

Unlimited: no limitations are in place.

Limited: there is a limit in place; you must enter the limit in the field below.

NoBandwidth: there is no bandwidth available.

Per call bandwidth limit (kbps)

Sets the limit on the bandwidth of individual calls via this pipe.

Applying pipes to links

Pipes are used to restrict the bandwidth of a link. When a pipe is applied to a link, it will restrict the bandwidth of calls made between the two nodes of the link - the restrictions will apply to calls in either direction.

Normally a single pipe would be applied to a single link. However, one or more pipes may be applied to one or more links, depending on how you wish to model your network.

One pipe, one link

Applying a single pipe to a single link is useful when you wish to apply specific limits to calls between a subzone and another specific subzone or zone.

One pipe, two or more links

Each pipe may be applied to multiple links. This is used to model the situation where one site communicates with several other sites over the same broadband connection to the Internet. A pipe should be configured to represent the broadband connection, and then applied to all the links. This will allow you to configure the bandwidth options for calls in and out of that site.

Example

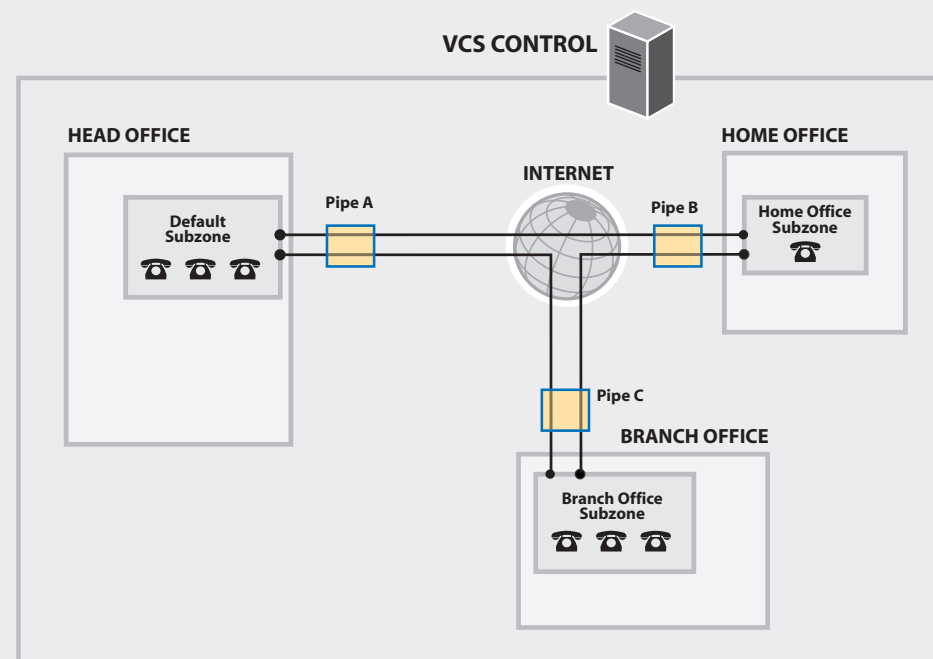
In the diagram opposite, Pipe A has been applied to two links: the link between the Default Subzone and the Home Office subzone, and the link between the Default Subzone and the Branch Office subzone. In this case, Pipe A represents the Head Office's broadband connection to the internet, and would have total and per-call restrictions placed on it.

Two pipes, one link

Each link may have up to two pipes associated with it. This is used to model the situation where the two nodes of a link are not directly connected, for example two sites that each have their own broadband connection to the Internet. Each connection should have its own pipe, meaning that a link between the two nodes should be subject to the bandwidth restrictions of both pipes.

Example

In the diagram opposite, the link between the Default Subzone and the Home Office Subzone has two pipes associated with it: Pipe A, which represents the Head Office's broadband connection to the internet, and Pipe B, which represents the Home Office's dial-up connection to the internet. Each pipe would have bandwidth restrictions placed on it to represent its maximum capacity, and a call placed via this link would have the lower of the two bandwidth restrictions applied.



Default bandwidth and downspeeding

About the default call bandwidth

Usually, when a call is initiated the endpoint will include in the request the amount of bandwidth it wishes to use. For those cases where the endpoint has not specified the bandwidth, you can set the VCS to apply a default bandwidth value.

About downspeeding

If bandwidth control is in use, there may be situations when there is insufficient bandwidth available to place a call at the requested rate. By default (and assuming that there is some bandwidth still available) the VCS will still attempt to connect the call, but at a reduced bandwidth – this is known as **downspeeding**.

Downspeeding can be configured so that it is applied in either or both of the following scenarios:

- when the requested bandwidth for the call exceeds the lowest **per-call** limit for the subzone or pipe(s)
- when placing the call at the requested bandwidth would mean that the **total** bandwidth limits for that subzone or pipe(s) would be exceeded.

You can turn off downspeeding, in which case if there is insufficient bandwidth to place the call at the originally requested rate, the call will not be placed at all. This could be used if, when your network is nearing capacity, you would rather a call failed to connect at all than be connected at a lower than requested speed. In this situation endpoint users will get one of the following messages, depending on the system that initiated the search:

- **Exceeds Call Capacity**
- **Gatekeeper Resources Unavailable**

Configuring default call bandwidth and downspeeding

To configure the default call bandwidth and downspeeding behavior using the web interface:

- **VCS configuration > Bandwidth > Configuration.**
You will be taken to the **Bandwidth Configuration** page.

To configure these settings using the CLI:

- [xConfiguration Bandwidth Default](#)
- [xConfiguration Bandwidth Downspeed](#)

The options are:

Default call bandwidth (kbps)

Enter the bandwidth value to be used for calls for which no bandwidth value has been specified by the system that initiated the call.



This value cannot be blank. The default value is 384kbps.

Downspeed per call mode

Determines what will happen if the **per-call** bandwidth restrictions on a subzone or pipe mean that there is insufficient bandwidth available to place a call at the requested rate.

On: the call will be downspeeded.

Off: the call will not be placed.

Downspeed total mode

Determines what will happen if the **total** bandwidth restrictions on a subzone or pipe mean that there is insufficient bandwidth available to place a call at the requested rate.

On: the call will be downspeeded.

Off: the call will not be placed.

Example without a firewall

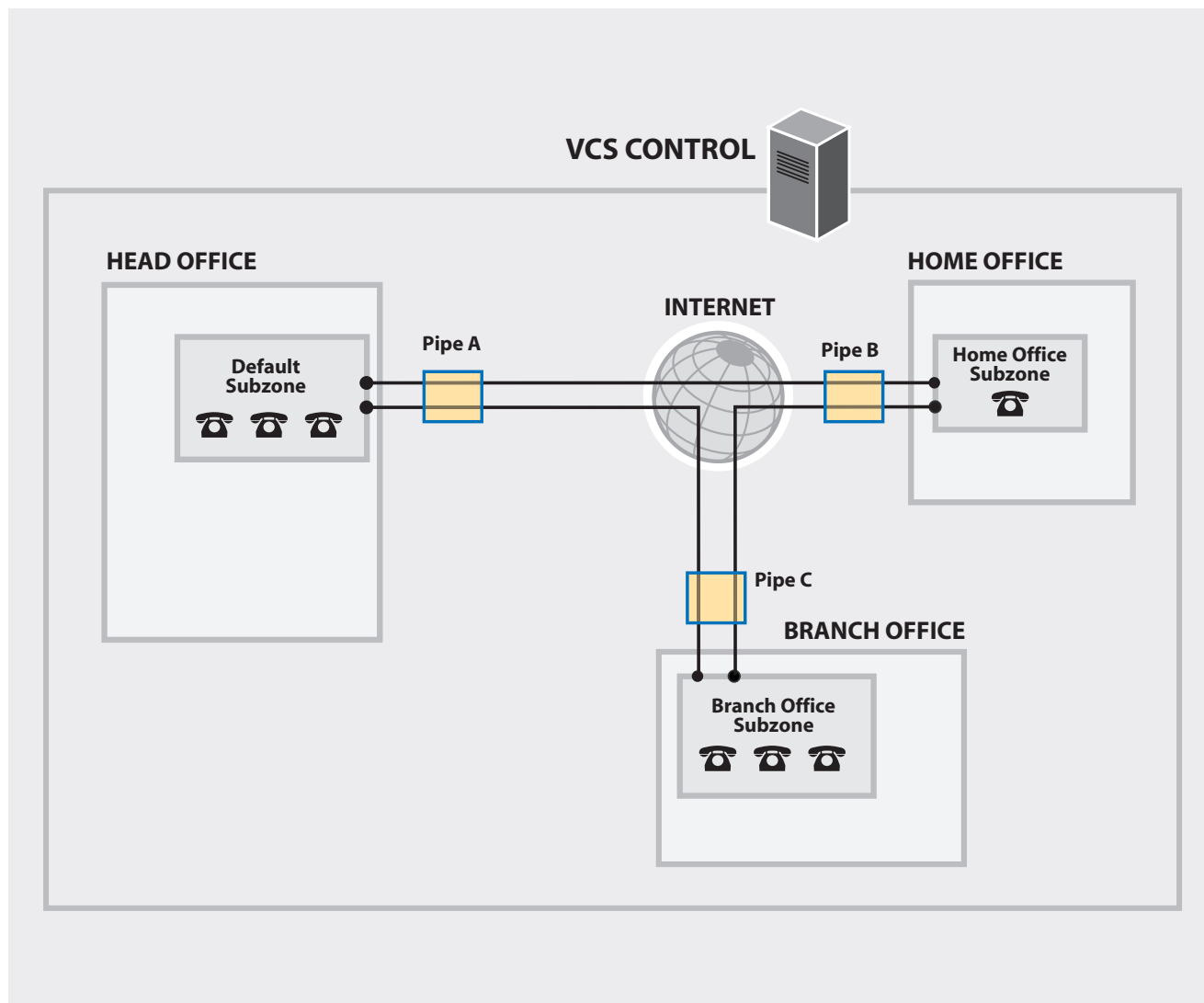
An example deployment is shown opposite. In this example, there are three geographically separate offices: Head, Branch and Home. All endpoints in the Head Office register with the VCS Control, as do those in the Branch and Home offices.

Each of the three offices is represented as a separate subzone on the VCS, with bandwidth configured according to local policy.

The enterprise's leased line connection to the Internet, and the DSL connections to the remote offices are modeled as separate pipes.

There are no firewalls involved in this scenario, so we can configure direct links between each of the offices. Each link is then assigned two pipes, representing the Internet connections of the offices at each end of the link.

In this scenario, a call placed between the Home Office and Branch Office will consume bandwidth from the Home and Branch subzones and on the Home and Branch pipes (Pipe B and Pipe C). The Head Office's bandwidth budget will be unaffected by the call.



Example with a firewall

If we modify the previous example deployment to include firewalls between the offices, we can use Cisco's Expressway™ firewall traversal solution to maintain connectivity. We do this by adding a VCS Expressway outside the firewall on the public internet, which will work in conjunction with the VCS Control and Home and Branch office endpoints to traverse the firewalls.

In this example, the endpoints in the Head Office register with the VCS Control, while those in the Branch and Home offices register with the VCS Expressway.

The introduction of the firewalls means that there is no longer any direct connectivity between the Branch and Home offices. All traffic must be routed through the VCS Expressway. This is shown by the absence of a link between the Home and Branch subzones.

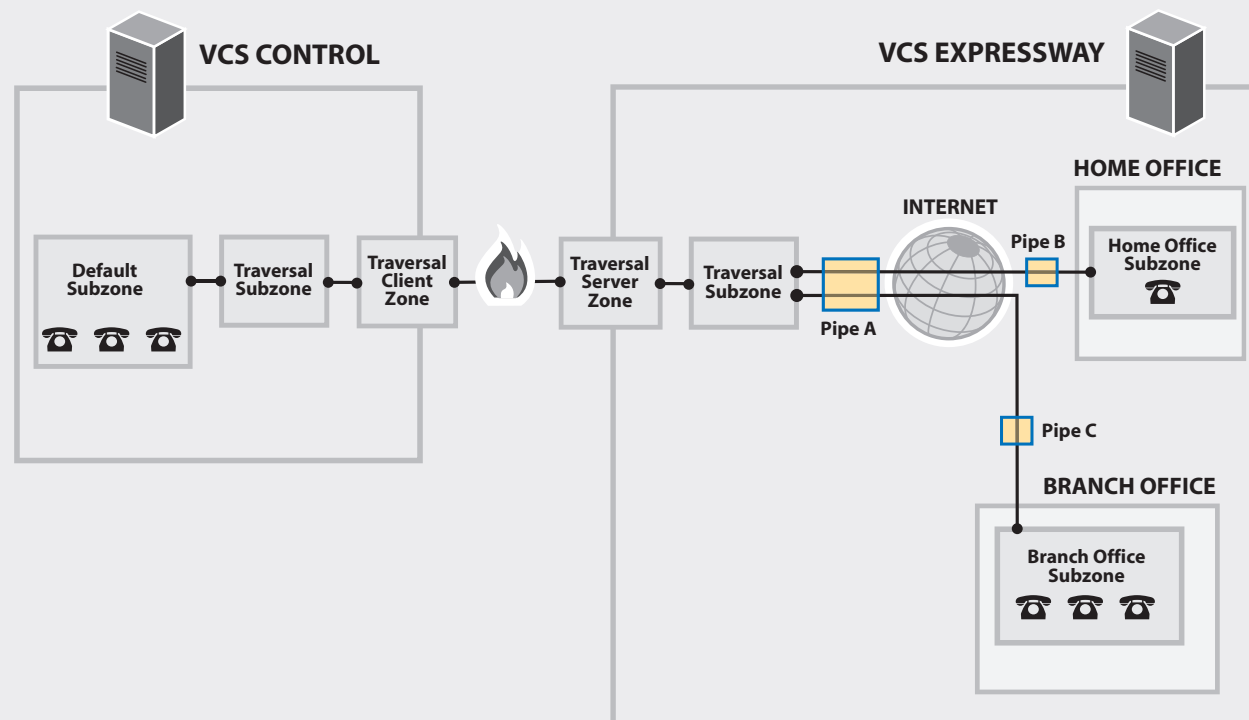
Cisco VCS Expressway subzone configuration

The VCS Expressway has subzones configured for the Home Office and Branch Office. These are linked to the VCS Expressway's Traversal Subzone, with pipes placed on each link. All calls from the VCS Expressway to the VCS Control must go through the Traversal Subzone and will consume bandwidth from this subzone. Note also that calls from the Home Office to the Branch Office must also go through the Traversal Subzone, and will also consume bandwidth from this subzone as well as the Home and Branch subzones and Home Office, Branch Office and Head Office pipes.

In this example we have assumed that there is no bottleneck on the link between the VCS Expressway and the Head Office network, so have not placed a pipe on this link. If you want to limit the amount of traffic flowing through your firewall, you could provision a pipe on this link.

Cisco VCS Control subzone configuration

Because the VCS Control is only managing endpoints on the Head Office LAN, its configuration is simpler. All of the endpoints in the Head Office are assigned to the Default Subzone. This is linked to the Traversal Subzone, through which all calls leaving the Head Office must pass.



Firewall traversal

This section describes how to configure your Cisco VCS Control and Cisco VCS Expressway in order to traverse firewalls. It also describes how to configure the additional firewall traversal server functions of a Cisco VCS Expressway, including TURN services.



About Expressway™

The purpose of a firewall is to control the IP traffic entering your network. Firewalls will generally block unsolicited incoming requests, meaning that any calls originating from outside your network will be prevented. However, firewalls can be configured to allow outgoing requests to certain trusted destinations, and to allow responses from those destinations. This principle is used by Cisco's Expressway™ solution to enable secure traversal of any firewall.

The Expressway™ solution consists of:

- a VCS Expressway or TANDBERG Border Controller located outside the firewall on the public network or in the DMZ, which acts as the firewall traversal server
- a VCS Control, TANDBERG Gatekeeper, MXP endpoint or other traversal-enabled endpoint located in a private network, which acts as the firewall traversal client

The two systems work together to create an environment where all connections between the two are outbound, i.e. established from the client to the server, and thus able to successfully traverse the firewall.

How does it work?

The traversal client constantly maintains a connection via the firewall to a designated port on the traversal server. This connection is kept alive by the client sending packets at regular intervals to the server. When the traversal server receives an incoming call for the traversal client, it uses this existing connection to send an incoming call request to the client. The client then initiates the necessary outbound connections required for the call media and/or signaling.

This process ensures that from the firewall's point of view, all connections are initiated from the traversal client inside the firewall out to the traversal server.



For firewall traversal to function correctly, the VCS Expressway must have one traversal server zone configured on it for each client system that is connecting to it (this does not include traversal-enabled endpoints which register directly with the VCS Expressway; the settings for these connections are configured in a different way). Likewise, each VCS client must have one traversal client zone configured on it for each server that it is connecting to. The ports and protocols configured for each pair of client-server zones must be the same. (See the [Quick guide to traversal client - server configuration](#) section for a summary of the required configuration on each system.) Because the VCS Expressway listens for connections from the client on a specific port, you are recommended to create the traversal server zone on the VCS Expressway before you create the traversal client zone on the VCS Control.

Cisco VCS as a firewall traversal client

Your VCS can act as a firewall traversal client on behalf of SIP and H.323 endpoints registered to it, and any gatekeepers that are neighbored with it.

In order to act as a firewall traversal client, the VCS must be configured with information about the system(s) that will be acting as its firewall traversal server. See the [Configuring the Cisco VCS as a traversal client](#) section for full details on how to do this.



In most cases, you will use a VCS Control as a firewall traversal client. However, a VCS Expressway can also act as a firewall traversal client.



The firewall traversal server used by the VCS client can be a VCS Expressway, or (for H.323 only) a TANDBERG Border Controller.

Cisco VCS as a firewall traversal server

The VCS Expressway has all the functionality of a VCS Control (including being able to act as a firewall traversal client). However, its main feature is that it can act as a firewall traversal server for other Cisco systems and any traversal-enabled endpoints that are registered directly to it. It can also provide TURN relay services to ICE-enabled endpoints. These features are enabled as follows:

- For the VCS Expressway to act as a firewall traversal server for Cisco systems, you must create and configure a new traversal server zone on the VCS Expressway for every system that is its traversal client. See the [Configuring the Cisco VCS as a traversal server](#) section for full instructions.
- For the VCS Expressway to act as a firewall traversal server for traversal-enabled endpoints (such as Cisco MXP endpoints and any other endpoints that support the ITU H.460.18 and H.460.19 standards), no additional configuration is required. See the [Configuring traversal for endpoints](#) section for more information.
- To enable TURN relay services and find out more about ICE, see the [TURN services](#) section.
- To reconfigure the default ports used by the VCS Expressway, see the [Configuring traversal server ports](#) section.

Quick guide to traversal client - server configuration

CISCO TELEPRESENCE VIDEO COMMUNICATION SERVER
ADMINISTRATOR GUIDE

Overview

Full details of how to configure a VCS Control and VCS Expressway as traversal client and server respectively are given in the following pages. However, the basic steps are:

- 1 On the VCS Expressway, create a traversal server zone (this represents the incoming connection from the VCS Control). In the **Client authentication username** field, enter the VCS Control's authentication username.
- 2 On the VCS Expressway, add the VCS Control's authentication username and password as credentials in the authentication database. These can be defined by going to **VCS configuration > Authentication > Devices > Local database**.
- 3 On the VCS Control, create a traversal client zone (this represents the connection to the VCS Expressway). Use the same **Authentication username** and **password** as specified on the VCS Expressway.
- 4 On the VCS Control, configure all the **modes** and **ports** in the H.323 and SIP protocol sections to match identically those of the traversal server zone on the VCS Expressway.
- 5 Enter the VCS Expressway's IP address or FQDN in the **Peer 1 address** field.

VCS Expressway (server)

1

Edit zone You are here: VCS.co

Configuration

Name * to Traversal Client 1 ⓘ

Type TraversalServer

Hop count * 15 ⓘ

Client authentication username * Client_Username ⓘ [Add/Edit local](#)

H.323

Mode On ⓘ

Protocol Assent ⓘ

Port * 6001 ⓘ

H.460.19 demultiplexing mode Off ⓘ

SIP

Mode On ⓘ

Port * 7001 ⓘ

Transport TLS ⓘ

TLS verify mode Off ⓘ

Accept proxied registrations Allow ⓘ

Poison mode Off ⓘ

2

Create credential You are here: VCS configuration > Authentication > Devices > Local

Configuration

Name * Client_Username ⓘ

Password * ⓘ

VCS Control (client)

3

Edit zone You are here: VCS.co

Configuration

Name * to Traversal Server ⓘ

Type TraversalClient

Hop count * 15 ⓘ

Authentication username * Client_Username ⓘ

Authentication password * ⓘ

H.323

Mode On ⓘ

Protocol Assent ⓘ

Port * 6001 ⓘ

SIP

Mode On ⓘ

Port * 7001 ⓘ

Transport TLS ⓘ

TLS verify mode Off ⓘ

Accept proxied registrations Allow ⓘ

Poison mode Off ⓘ

Client settings

Retry interval * 120 ⓘ

Location

Peer 1 address * traversalserver@example.com ⓘ

Peer 2 address ⓘ

4

5

Overview

Ports play a vital part in firewall traversal configuration. The correct ports must be set on the VCS Expressway, traversal client and firewall in order for connections to be permitted.

Ports are initially configured on the VCS Expressway by the VCS Expressway administrator. The firewall administrator and the traversal client administrator should then be notified of the ports, and they must then configure their systems to connect to these specific ports on the server. The only port configuration that is done on the client is the range of ports it uses for outgoing connections; the firewall administrator may need to know this information so that if necessary they can configure the firewall to allow outgoing connections from those ports.



The pages under the **Maintenance > Tools > Port usage** menu show, in table format, all the IP ports that are being used on the VCS, both inbound and outbound. This information can be provided to your firewall administrator so that the firewall can be configured appropriately. See the [Port usage](#) section for further information.

Expressway process

The Expressway™ solution works as follows:

1. Each traversal client connects via the firewall to a unique port on the VCS Expressway.
2. The server identifies each client by the port on which it receives the connection, and the authentication credentials provided by the client.
3. After the connection has been established, the client constantly sends a probe to the VCS Expressway via this connection in order to keep the connection alive.
4. When the VCS Expressway receives an incoming call for the client, it uses this initial connection to send an incoming call request to the client.
5. The client then initiates one or more outbound connections. The destination ports used for these connections will differ for signaling and/or media, and will depend on the protocol being used (see the following sections for more details).

H.323 firewall traversal protocols

The VCS supports two different firewall traversal protocols for H.323: Assent and H.460.18/H.460.19.

- Assent is Cisco's proprietary protocol.
- H.460.18 and H.460.19 are ITU standards which define protocols for the firewall traversal of signaling and media respectively. These standards are based on the original Assent protocol.

A traversal server and traversal client must use the same protocol in order to communicate.

The two protocols each use a different range of ports.

SIP firewall traversal protocols

The VCS supports the Assent protocol for SIP firewall traversal of media.

The signaling is traversed through a TCP/TLS connection established from the client to the server.

Ports for initial connections from traversal clients

Each traversal server zone specifies an **H.323 port** and a **SIP port** to be used for the initial connection from the client.

Each time you configure a new traversal server zone on the VCS Expressway, you will be allocated default port numbers for these connections:

- H.323 ports start at UDP/6001 and increment by 1 for every new traversal server zone
- SIP ports start at TCP/7001 and increment by 1 for every new traversal server zone.

You can change these default ports if necessary but you must ensure that the ports are unique for each traversal server zone.

After the H.323 and SIP ports have been set on the VCS Expressway, matching ports must be configured on the corresponding traversal client.



The default port used for the initial connections from MXP endpoints is the same as that used for standard RAS messages, i.e. UDP/1719. While it is possible to change this port on the VCS Expressway, most endpoints will not support connections to ports other than UDP/1719. You are therefore recommended to leave this as the default.



You must allow outbound connections through your firewall to each of the unique SIP and H.323 ports that are configured on each of the VCS Expressway's traversal server zones.

Assent ports

For connections to the VCS Expressway using the **Assent** protocol, the default ports are:

Call signaling

- UDP/1719: listening port for RAS messages
- TCP/2776: listening port for H.225 and H.245 protocols

Media

- UDP/2776: RTP media port
- UDP/2777: RTCP media control port



If your VCS Expressway does not have any endpoints registering directly with it, and it is not part of a cluster, then UDP/1719 is not required. You therefore do not need to allow outbound connections to this port through the firewall between the VCS Control and VCS Expressway.

H.460.18/19 ports

For connections to the VCS Expressway using the **H.460.18/19** protocols, the default ports are:

Call signaling

- UDP/1719: listening port for RAS messages
- TCP/1720: listening port for H.225 protocol
- TCP/2777: listening port for H.245 protocol

Media

- UDP/2776: RTP media port
- UDP/2777: RTCP media control port
- UDP/50000-52399: demultiplex media port range

Ports for connections out to the public internet

In situations where the VCS Expressway is attempting to connect to an endpoint on the public internet, you will not know the exact ports on the endpoint to which the connection will be made. This is because the ports to be used are determined by the endpoint and advised to the VCS Expressway only after the server has located the endpoint on the public internet. This may cause problems if your VCS Expressway is located within a DMZ (i.e. there is a firewall between the VCS Expressway and the public internet) as you will not be able to specify in advance rules that will allow you to connect out to the endpoint's ports.

You can however specify the ports on the VCS Expressway that will be used for calls to and from endpoints on the public internet so that your firewall administrator can allow connections via these ports. The ports that can be configured for this purpose are:

H.323

- TCP/1720: signaling
- UDP/1719: signaling
- UDP/50000-52399: media
- TCP/15000-19999: signaling

SIP

- TCP/5061: signaling
- UDP/5060 (default): signaling
- UDP/50000-52399: media
- TCP: a temporary port in the range 25000-29999 is allocated.

TURN

- UDP/3478 (default): TURN services
- UDP/60000-61200 (default range): media

SIP ports

Call signaling

SIP call signaling uses the same port as used by the initial connection between the client and server.

Media

Where the traversal client is a VCS, SIP media uses Assent to traverse the firewall. The default ports are the same as for H.323, i.e.:

- UDP/2776: RTP media port
- UDP/2777: RTCP media control port

TURN ports

The VCS Expressway can be enabled to provide [TURN services](#) (Traversal Using Relays around NAT) which can be used by SIP endpoints that support the [ICE firewall traversal protocol](#).

The ports used by these services are configurable using:

- **VCS configuration > Expressway > TURN**
- [xConfiguration Traversal Server TURN](#)

The ICE clients on each of the SIP endpoints must be able to discover these ports, either by using SRV records in DNS or by direct configuration.

Overview

To control which systems can use the VCS Expressway as a traversal server, each VCS Control or Gatekeeper that wants to be its client must first authenticate with it.

Upon receiving the initial connection request from the traversal client, the VCS Expressway asks the client to authenticate itself by providing its authentication credentials. The VCS Expressway then looks up the client's credentials in its own authentication database. If a match is found, the VCS Expressway accepts the request from the client.

The settings used for authentication depend on the combination of client and server being used. These are detailed in the table opposite.



All VCS and Gatekeeper traversal clients must authenticate with the VCS Expressway, regardless of the VCS Expressway's **Authentication mode** setting. However, endpoint clients are only required to authenticate if the VCS Expressway's **Authentication mode** is **On**.

Authentication and NTP

All VCS and Gatekeeper traversal clients that support H.323 must authenticate with the VCS Expressway. The authentication process makes use of timestamps and requires that each system uses an accurate system time. The system time on a VCS is provided by a remote NTP server. Therefore, for firewall traversal to work, all systems involved must be [configured with details of an NTP server](#).

Client	Server
Cisco VCS Control or Cisco VCS Expressway The VCS client provides its Authentication username and Authentication password. These are set on the traversal client zone by using VCS configuration > Zones > Edit zone, in the Configuration section.	Cisco VCS Expressway - The traversal server zone for the VCS client must be configured with the Client authentication username. This is set on the VCS Expressway by using VCS configuration > Zones > Edit zone, in the Configuration section. - There must also be an entry in the VCS Expressway's authentication database with the corresponding client username and password.
Endpoint The endpoint client provides its Authentication ID and Authentication Password.	Cisco VCS Expressway There must be an entry in the VCS Expressway's authentication database with the corresponding client username and password.
TANDBERG Gatekeeper (version 5.2 and earlier) The Gatekeeper looks up its System Name in its own authentication database and retrieves the password for that name. It then provides this name and password.	Cisco VCS Expressway - The traversal server zone for the Gatekeeper client must be configured with the Gatekeeper's System Name in the Client authentication username field. This is set on the VCS Expressway by using VCS configuration > Zones > Edit zone, in the Configuration section. - There must be an entry in the VCS Expressway's authentication database that has the Gatekeeper's System name as the username, along with the corresponding password.
TANDBERG Gatekeeper (version 6.0 or later; 6.1 or later is the recommended version) The Gatekeeper provides its Authentication Username and Authentication Password. These are set on the Gatekeeper by using Gatekeeper Configuration > Authentication, in the External Registration Credentials section.	Cisco VCS Expressway - The traversal server zone for the Gatekeeper client must be configured with the Gatekeeper's Authentication Username. This is set on the VCS Expressway by using VCS configuration > Zones > Edit zone, in the Configuration section. - There must also be an entry in the VCS Expressway's authentication database with the corresponding client username and password.
Cisco VCS Control or Cisco VCS Expressway - If Authentication is On on the Border Controller, the VCS client provides its Authentication username and Authentication password. These are set on the traversal client zone by using VCS configuration > Zones > Edit zone, in the Configuration section. - If the Border Controller is in Assent mode, the VCS client provides its Authentication username. This is set on the traversal client zone by using VCS configuration > Zones > Edit zone, in the Configuration section.	TANDBERG Border Controller - If Authentication is On on the Border Controller, there must be an entry in the Border Controller's authentication database that matches the VCS client's Authentication username and Authentication password. - If the Border Controller is in Assent mode, the traversal zone configured on the Border Controller to represent the VCS client must use the VCS's Authentication username in the Assent Account name field. This is set on the Border Controller via TraversalZone > Assent > Account name.

Firewall traversal and Dual Network Interfaces

The Dual Network Interfaces option key enables the LAN 2 interface on your VCS Expressway (the option is not available on a VCS Control). The LAN 2 interface is used in situations where your VCS Expressway is located in a DMZ that consists of two separate networks - an inner DMZ and an outer DMZ - and your network is configured to prevent direct communication between the two.

With the LAN 2 interface enabled, you can configure the VCS with two separate IP addresses, one for each network in the DMZ. Your VCS then acts as a proxy server between the two networks, allowing calls to pass between the internal and outer firewalls that make up your DMZ.



All ports configured on the VCS, including those relating to firewall traversal, apply to both IP addresses; it is not possible to configure these ports separately for each IP address.

Firewall configuration

In order for Expressway™ firewall traversal to function correctly, the firewall must be configured to:

- allow initial outbound traffic from the client to the ports being used by the VCS Expressway
- allow return traffic from those ports on the VCS Expressway back to the originating client

Cisco offers a downloadable tool, the Expressway Port Tester, that allows you to test your firewall configuration for compatibility issues with your network and endpoints. It will advise if necessary which ports may need to be opened on your firewall in order for the Expressway™ solution to function correctly. The Expressway Port Tester currently only supports H.323. Contact your Cisco representative for more information.



You are recommended to turn off any H.323 and SIP protocol support on the firewall: these are not needed in conjunction with the Cisco Expressway™ solution and may interfere with its operation.



The pages under the **Maintenance > Tools > Port usage** menu show, in table format, all the IP ports that are being used on the VCS, both inbound and outbound. This information can be provided to your Firewall Administrator in order to allow them to configure the firewall appropriately. See the [Port usage](#) section for further information.

Adding and configuring a traversal client zone

To enable your VCS to act as a traversal client on behalf of its endpoints and neighbor gatekeepers, you must create a connection between it and a traversal server (e.g. a VCS Expressway or Border Controller).

You do this by adding a new traversal client zone on the VCS client and configuring it with the details of the traversal server.

To add a new traversal client zone:

- **VCS configuration > Zones.**
You are taken to the **Zones** page.
Click **New**.
You are taken to the **Create zone** page.
- [xCommand ZoneAdd](#)

To edit an existing traversal client zone:

- **VCS configuration > Zones.**
You are taken to the **Zones** page.
Click on the name of the zone you want to configure.
You are taken to the **Edit zone** page.
- [xConfiguration Zones Zone \[1..1000\]](#)
- [xConfiguration Zones Zone \[1..1000\] Traversal Client](#)

For details on the configuration options available for traversal client zones, see the [Configuring traversal client zones](#) section.



You can create more than one traversal client zone if you want to connect to multiple traversal servers.

Configuring the Cisco VCS as a traversal server

CISCO TELEPRESENCE VIDEO COMMUNICATION SERVER
ADMINISTRATOR GUIDE

Overview

The VCS Expressway can act as a firewall traversal server. This feature means you can:

- Allow your VCS to act as a traversal server for other VCSs and TANDBERG Gatekeepers. You do this by adding a new traversal server zone on the VCS, and configuring it with details of the traversal client.
- Provide firewall traversal for any traversal-enabled endpoints (i.e. Cisco MXP endpoints and any other endpoints that support the ITU H.460.18 and H.460.19 standards) registered directly with it. You can configure the protocols and ports that will be used.
- Enable and configure TURN services.
- Configure the ports used specifically for firewall traversal services.

The following sections describe how to configure each of the above options.

Adding and configuring a traversal server zone

To enable your VCS Expressway to act as a traversal server for a traversal client (e.g. a VCS Control or Gatekeeper) you must create a connection between it and the client system.

You do this by adding a new traversal server zone on the VCS server and configuring it with the details of the traversal client.

To add a new traversal server zone:

- **VCS configuration > Zones.**
You are taken to the **Zones** page. Click **New**.
You are taken to the **Create zone** page.
- [xCommand ZoneAdd](#)

To edit an existing traversal server zone:

- **VCS configuration > Zones.**
You are taken to the **Zones** page. Click on the name of the zone you want to configure.
You are taken to the **Edit zone** page.
- [xConfiguration Zones Zone](#)
- [xConfiguration Zones Zone \[1..1000\] TraversalServer](#)

For details on the configuration options available for traversal server zones, see the [Configuring traversal server zones](#) section.

Configuring traversal for endpoints

Overview

Traversal-enabled H.323 endpoints can register directly with the VCS Expressway and use it for firewall traversal.

To configure the options for these endpoints:

- **VCS configuration > Expressway> Locally registered endpoints**
You are taken to the **Locally registered endpoints** page.
- [xConfiguration Zones LocalZone Traversal H323](#)

The options available are:

H.323 Assent mode

Determines whether or not H.323 calls using Assent mode for firewall traversal are allowed.

H.460.18 mode

Determines whether or not H.323 calls using H.460.18/19 mode for firewall traversal are allowed.

H.460.19 demux mode

Determines whether the VCS Expressway operates in demultiplexing mode for calls from locally registered endpoints.

On: allows use of the same two ports for all calls.

Off: each call will use a separate pair of ports for media.

H.323 preference

If an endpoint supports both Assent and H.460.18 protocols, this setting determines which the VCS Expressway uses.

UDP probe retry interval

Sets the frequency (in seconds) with which locally registered endpoints send a UDP probe to the VCS Expressway.

UDP probe retry count

Sets the number of times locally registered endpoints attempt to send a UDP probe to the VCS Expressway.

UDP probe keep alive interval

Sets the interval (in seconds) with which locally registered endpoints send a UDP probe to the VCS Expressway after a call is established, in order to keep the firewall's NAT bindings open.

TCP probe retry interval

Sets the frequency (in seconds) with which locally registered endpoints send a TCP probe to the VCS Expressway.

TCP probe retry count

Sets the number of times locally registered endpoints attempt to send a TCP probe to the VCS Expressway.

TCP probe keep alive interval

Sets the interval (in seconds) with which locally registered endpoints send a TCP probe to the VCS Expressway after a call is established, in order to keep the firewall's NAT bindings open.

Configuring traversal server ports

Overview

The VCS Expressway has specific listening ports used for firewall traversal. Rules must be set on your firewall to allow connections to these ports. In most cases the default ports should be used. However, you have the option to change these ports if necessary.

Configuration

To configure the VCS Expressway ports:

- **VCS configuration > Expressway > Ports**
You are taken to the **Ports** page.
- [xConfiguration Traversal Server Media Demultiplexing](#)
- [xConfiguration Traversal Server H.323](#)

The options are:

Media demultiplexing RTP port

Specifies the port on the VCS Expressway used for demultiplexing RTP media.

Media demultiplexing RTCP port

Specifies the port on the VCS Expressway used for demultiplexing RTCP media.

H.323 Assent call signaling port

Specifies the port on the VCS Expressway used for Assent signaling.

H.323 H.460.18 call signaling port

Specifies the port on the VCS Expressway used for H.460.18 signaling.

TURN services

About ICE

ICE (Interactive Connectivity Establishment) provides a mechanism for SIP client NAT traversal. ICE is not a protocol, but a framework which pulls together a number of different techniques such as TURN and STUN.

It allows endpoints (clients) residing behind NAT devices to discover paths through which they can pass media, verify peer-to-peer connectivity via each of these paths and then select the optimum media connection path. The available paths typically depend on any inbound and outbound connection restrictions that have been configured on the NAT device. Such behavior is described in [RFC 4787 \[13\]](#).

An example usage of ICE is two home workers communicating via the internet. If the two endpoints can communicate via ICE the VCS Expressway may (depending on how the NAT devices are configured) only need to take the signaling and not take the media (and is therefore a non-traversal call). If the initiating ICE client attempts to call a non-ICE client, the call set-up process reverts to a conventional SIP call requiring NAT traversal via media latching where the VCS also takes the media and thus requires a traversal licence.

About TURN

TURN (Traversal Using Relays around NAT) services are relay extensions to the STUN network protocol that enable a SIP or H.323 client to communicate via UDP or TCP from behind a NAT device. Currently the VCS supports TURN over UDP only.



For detailed information on the base STUN protocol, refer to [Session Traversal Utilities for \(NAT\) \(STUN\) \[11\]](#).

TURN relay server

The VCS Expressway's TURN relay server can be configured to provide TURN services to traversal clients.

How TURN is used by an ICE client

Each ICE client requests the TURN server to allocate relays for the media components of the call. A relay is required for each component in the media stream between each client.

After the relays are allocated, each ICE client has 3 potential connection paths (addresses) through which it can send and receive media:

- its host address which is behind the NAT device (and thus not reachable from endpoints on the other side of the NAT)
- its publicly-accessible address on the NAT device
- a relay address on the TURN server

The endpoints then decide, by performing connectivity checks through ICE, how they are going to communicate. Depending upon how the NAT devices are configured, the endpoints may be able to communicate between their public-facing addresses on the NAT devices or they may have to relay the media via the TURN server. If both endpoints are behind the same NAT device they can send media directly between themselves using their internal host addresses.

After the media route has been selected the TURN relay allocations are released if the chosen connection paths do not involve routing via the TURN server. Note that the signaling always goes via the VCS, regardless of the final media communication path chosen by the endpoints.

Capabilities and limitations

- The VCS supports up to 70 relay allocations. This is typically enough to support 5 calls but does depend on the network topology and the number of media stream components used for the call (for example, some calls may use Duo Video, or other calls might be audio only).
- Clustered VCSs: if the requested TURN server's relays are fully allocated the server will respond to the requesting client with the details of an alternative server in the cluster (the TURN server currently with the most available resources).
- The VCS's TURN services are supported over single and dual network interfaces. For dual network interfaces, relays are allocated on the VCS's externally facing LAN interface.
- ICE calls can only be made between devices registered to the VCS's Local Zone.
- Microsoft ICE (which is not standards-based) is not supported.
- The TURN server does not support bandwidth requests. (Note that traversal zone bandwidth limits do not apply.)

TURN relay status information

The [TURN relays](#) page (**Status > TURN relays**) lists all the currently active TURN relays on the VCS.

You can also review further details of each TURN relay including permissions, channel bindings and counters.



For detailed information on the TURN relay service, refer to [Traversal Using Relays around NAT \(TURN\) \[12\]](#).

Configuring TURN services

TURN relay services are only available on a VCS Expressway. To use TURN services you also need the TURN Relay option key (this controls the number of TURN relays that can be simultaneously allocated by the VCS).

To configure the VCS's TURN services:

- **VCS configuration > Expressway > TURN**
You are taken to the **TURN** page.
- [xConfiguration Traversal Server TURN](#)

The configurable options are:

TURN services

Determines whether the VCS offers TURN services to traversal clients.

Port

The listening port for TURN requests. The default is 3478.

Authentication realm

The realm sent by the server in its authentication challenges.



Ensure the client's credentials are stored in the [device authentication](#) database.

Media port range start / end

The lower and upper port in the range used for the allocation of TURN relays.

Applications

This section provides information on each of the additional services that are available under the **Applications** menu of the Cisco VCS. You may need to purchase the appropriate option key in order to use each of these applications. They are:

- [Conference Factory](#)
- [Presence services](#)
- [OCS Relay](#)
- [FindMe](#)
- [Provisioning \(Starter Pack\)](#)

This section also provides information on:

- [TMS Agent](#)



Overview

The Conference Factory application allows the VCS to support the Multiway™ feature. Multiway enables endpoint users to create a conference while in a call even if their endpoint does not have this functionality built in. Multiway is supported in Cisco TelePresence endpoints including the E20 (software version TE1.0 or later) and MXP range (software version F8.0 or later).

Check with your Cisco representative for an up-to-date list of the endpoints and infrastructure products that support Multiway.

Process

When the Multiway feature is activated from the endpoint:

1. The endpoint calls a pre-configured alias which routes to the Conference Factory on the VCS.
2. The VCS replies to the endpoint with the alias that the endpoint should use for the Multiway conference. This alias will route to an MCU.
3. The endpoint then places the call to the MCU using the given alias, and informs the other participating endpoints to do the same.



Refer to the [VCS Multiway Deployment Guide \[25\]](#) for full details on how to configure individual components of your network (endpoints, MCUs and VCSs) in order to use Multiway in your deployment.

Configuration

The Conference Factory application is enabled and configured using the **Conference Factory** page (**Applications > Conference Factory**).

Mode

The **Mode** option allows you to enable or disable the Conference Factory application.

Alias

The alias that will be dialed by the endpoints when the Multiway feature is activated. This must also be configured on all endpoints that may be used to initiate the Multiway feature. An example could be *multiway@example.com*.

Template

The alias that the VCS tells the endpoint to dial to create a Multiway conference on the MCU.

To ensure that each conference has a different alias, you should use %% as a part of the template. The %% will be replaced by a unique number each time the VCS receives a new conference request.

Number range start / end

The first and last numbers in the range that replaces %% in the template used to generate a conference alias.

For example, your **Template** could be *563%%@example.com* with a range of 10 - 999. The first conference will use the alias *563010@example.com*, the next conference will use *563011@example.com* and so on up to *563999@example.com*., after which it will loop round and start again at *563010@example.com*. (Note that the %% represents a fixed number of digits based upon the upper range limit.)



You must use a different **Template** on each VCS in your network that has the Conference Factory application enabled. If your VCS is part of a cluster, the **Template** must be different for each peer in the cluster.



The alias generated by the **Template** must be a fully-qualified SIP alias and must route to the MCU. The MCU must be configured to process this alias. No other special configuration is required on the MCU in order to support the Conference Factory application.



SIP mode must be set to **On** (**VCS configuration > Protocols > SIP > Configuration**) for the Conference Factory application to function. If you want to be able to initiate calls to the Conference Factory from H.323 endpoints, you must also set **H.323 mode** to **On** (**VCS configuration > Protocols > H.323**), and ensure that **H.323 <-> SIP interworking mode** is set to **RegisteredOnly** or **On** (**VCS configuration > Protocols > Interworking**).

Overview

Presence is the ability of endpoints to provide information to other users about their current status - such as whether they are offline, online, or in a call. Any entity which provides presence information, or about whom presence information can be requested, is known as a presentity. Presentities publish information about their own presence status, and also subscribe to the information being published by other presentities and FindMe users.

Endpoints that support presence, such as Movi™ v2.0 (or later) clients, can publish their own status information. The VCS can also provide basic presence information on behalf of endpoints that do not support presence, including H.323 endpoints, as long as they have registered with an alias in the form of a URI.

If FindMe is enabled, the VCS can also provide presence information about FindMe users by aggregating the information provided by each presentity configured for that FindMe user.

The Presence application on the VCS supports the SIP-based SIMPLE standard and is made up of two separate services. These are the [Presence Server](#) and the [Presence User Agent](#) (PUA). These services can be [enabled and disabled](#) separately.

The [Presence status pages](#) provide information about the presentities who are providing presence information and the users who are requesting presence information on others.



Presence is supported by clustering. For specific information about how Presence information is managed across peers in a cluster, see the [Clustering and Presence](#) section.

Presence Server

The Presence Server application on the VCS is responsible for managing the presence information for all presentities in the SIP domain(s) for which the VCS is authoritative (refer to the [SIP domains](#) section for more information). The Presence Server can manage the presence information for locally registered endpoints and presentities whose information has been received via a SIP proxy (e.g. another VCS Control or Expressway).

The Presence Server is made up of the following services, all of which are enabled (or disabled) simultaneously when the Presence Server is enabled (or disabled):

- **Publication Manager:** receives PUBLISH messages, which contain the status information about a presentity, and writes this information to the Presence Database. PUBLISH messages are generated by presence-enabled endpoints and by the [Presence User Agent](#) (PUA).
- **Subscription Manager:** handles SUBSCRIBE messages, which request information about the status of a presentity. Upon receipt of a SUBSCRIBE message, the Subscription Manager sends a request to the Presentity Manager for information about that presentity, and forwards the information that is returned to the subscriber. The Subscription Manager also receives notifications from the Presentity Manager when a presentity's status has changed, and send this information to all subscribers.
- **Presentity Manager:** an interface to the Presence Database. It is used to support VCS features such as FindMe and the PUA, where the presence information provided by a number of different devices must be aggregated in order to provide an overall presence status for one particular presentity. When the Presentity Manager receives a request from the subscription manager for information on a presentity, it queries the Presence Database for all information available on all the endpoints associated with that particular presentity. The Presentity Manager then aggregates this information to determine the presentity's current status, and returns this to the Subscription Manager.
- **Presence Database:** stores current presence information received in the form of PUBLISH messages. Also sends NOTIFY messages to the Presentity Manager to inform it of any changes.

Presence User Agent (PUA)

Overview

Endpoints that do not support presence can have status published on their behalf by the VCS. The service that publishes this information is called the Presence User Agent (PUA).

The PUA takes information from the local registration database and the call manager and determines, for each endpoint that is currently locally registered, whether or not it is currently in a call. The PUA then provides this status information via a PUBLISH message.

In order for the PUA to successfully provide presence information about a locally registered endpoint:

- the endpoint must be registered with an alias in the form of a URI
- the domain part of the URI must be able to be routed to a SIP registrar that has a presence server enabled. (This could be either the local Presence Server, if enabled, or another Presence Server on a remote system.)

When enabled, the PUA generates presence information for all endpoints registered to the VCS, including those which already support presence. The status information provided by the PUA is either:

- **online** (registered but not in a call)
- **in call** (registered and currently in a call)

Aggregation of presence information

When enabled, the PUA generates presence information for all endpoints registered to the VCS, including those which already support presence. However, endpoints that support presence may provide other, more detailed status, for example **away** or **do not disturb**. For this reason, information provided by the PUA is used by the Presentity Manager as follows:

- Where presence information is provided by the PUA and one other source, the non-PUA presence information will always be used in preference to the PUA presence information. This is because it is assumed that the other source of information is the presentity itself, and this information is more accurate.
- Where presence information is provided by the PUA and two or more other sources, the Presence Server will aggregate the presence information from all presentities to give the 'highest interest' information, e.g. **online** rather than **offline**, and **in call** rather than **away**.
- If no information is being published about an endpoint, either by the endpoint itself or by the PUA, the endpoint's status will be **offline**. If the PUA is enabled, the **offline** status indicates that the endpoint is not currently registered.

FindMe presence

When the Presentity Manager receives a request for information about the presences of a FindMe alias, it looks up the presence information for each endpoint that makes up that FindMe alias. It then aggregates this information as follows:

- if the FindMe alias is set to **Individual** mode, if any one of the endpoints making up that FindMe is in a call the FindMe presentity's status will be reported as **in call**.
- if the FindMe alias is set to **Group** mode, if any one of the endpoints is online (i.e. not in call or offline) then the FindMe presentity's status will be reported as **online**.

Registration refresh period

The PUA will update and publish presence information on receipt of:

- a registration request (for new registrations)
- a registration refresh (for existing registrations)
- a deregistration request
- call setup and teardown information

For non-traversal H.323 registrations the default registration refresh period is 30 minutes. This means that when the PUA is enabled on a VCS with existing registrations, it may take up to 30 minutes before an H.323 registration refresh is received and **available** presence information is published for that endpoint. It also means that if an H.323 endpoint becomes unavailable without sending a deregistration message, it may take up to 30 minutes for its status to change to **offline**. To ensure more timely publication of presence information for H.323 endpoints, you should decrease the H.323 registration refresh period (using **VCS configuration > Protocols > H.323 > Gatekeeper > Time to live**).

The default registration refresh period for SIP is 60 seconds, so it will take no more than a minute for the PUA to publish updated presence information on behalf of any SIP endpoints.

Configuring Presence

Enabling and disabling Presence Services

Presence Services (the Presence Server and the Presence User Agent) are both disabled by default.

These services can be enabled and disabled separately from each other, depending on the nature of your deployment.

To enable and disable the [Presence Server](#) and [Presence User Agent](#):

- Go to **Applications > Presence**
You will be taken to the **Presence** page.
- [xConfiguration Applications Presence](#)

Presence User Agent (PUA)

Enabled

If the PUA is enabled, it will publish presence information for all locally registered endpoints, whether or not those endpoints are also publishing their own presence information. Information published by the PUA will be routed to a Presence Server acting for the endpoint's domain. This could be the local Presence Server, or (if this is disabled) a Presence Server on another system that is authoritative for that domain.

Disabled

If the PUA is disabled, only those endpoints that support presence will publish presence information. No information will be available for endpoints that do not support presence.

Presence Server

Regardless of whether or not the Presence Server is enabled, the VCS will still continue to receive PUBLISH messages if they are sent to it from any of the following sources:

- locally registered endpoints that support presence
- the local PUA (if enabled)
- remote SIP Proxies

Enabled

If the local Presence Server is enabled, it will process any PUBLISH messages intended for the SIP domains for which the

local VCS is authoritative. All other PUBLISH messages will be proxied on in accordance with the VCS's SIP routing rules.



SIP routes are configured using the CLI only. See [xConfiguration SIP Routes Route \[1..20\]](#) for details.

Disabled

If the local Presence Server is disabled, the VCS will proxy on all PUBLISH messages to one or more of its neighbor zones in accordance with its locally configured [call processing](#) rules. The local VCS will do this regardless of whether or not it is authoritative for the presentity's domain. If one of these neighbors is authoritative for the domain, and has a Presence Server enabled, then that neighbor will provide presence information for the presentity.

Recommendations

VCS Expressway and VCS Control

The recommended configuration for a VCS Expressway when acting as a traversal server for a VCS Control is to enable the PUA and disable the Presence Server on the VCS Expressway, and enable the Presence Server on the VCS Control. This will ensure that all PUBLISH messages generated by the PUA are routed to the VCS Control.

VCS neighbors

We recommend that if you have a deployment with two or more VCSs neighbored together, you enable the presence server on just one VCS. This will ensure a central source of information for all presentities in your network.

VCS clusters

For information about how Presence works within a VCS cluster, see the [Clustering and Presence](#) section.



Note that any defined [transforms](#) also apply to any Publication, Subscription or Notify URIs handled by the Presence Services.

Presence Server expiration times

Subscription expiration time

This is the maximum time (in seconds) within which a subscriber must refresh its subscription. If the subscriber does not send a refresh within this period, the Presence Server will stop sending NOTIFY messages to it.

You may wish to increase this value in deployments with large numbers of endpoints, to prevent too many messages being sent over your network.

Publication expiration time

This is the maximum time (in seconds) within which a publisher must refresh its publication. If the publisher does not send a refresh within this period, the Presence Server will show its presence as **Offline**.

You may wish to increase this value in deployments with large numbers of endpoints, to prevent too many messages being sent over your network.

Viewing presence status

Publishers

Status > Applications > Presence > Publishers

This page lists each presentity whose presence information is being managed by (i.e. published to) the local Presence Server. All Presentities are listed here regardless of whether or not anyone is requesting their presence information. If there are no Publishers listed, this could mean that the Presence Server is not enabled on this VCS.

Note: FindMe users are not listed here as they do not have their status individually published. The status of a FindMe user is based on the published status of the endpoints and/or presentities that make up the FindMe user, and is determined by the Presentity Manager.

URI: The address of the presentity whose presence information is being published.

Document count: The number of sources of information that are being published for this particular presentity. All endpoints that are registered to the VCS will have information published on their behalf by the PUA (as long as they are registered with an alias in the form of a URI). If an endpoint supports presence, it may also publish its own presence information. This means that some presentities will have more than one source of information about their presence. It is the job of the Presentity Manager to aggregate this information and determine the actual status of the presentity.

Presentities

Status > Applications > Presence > Presentities

This page lists each presentity whose presence information is being managed by (i.e. published to) the local Presence Server and whose presence information has been requested by a subscriber. Presentities are listed here whether or not there is any information currently available about that presentity. If a presentity has been subscribed to but there is no information being published about it, then it will be listed here if the local presence server is authoritative for the presentity's domain. Presentities are listed here regardless of whether the subscriber that requested the information is registered locally or to a remote system.

Note: FindMe users will be listed here if their presence information has been requested by a subscriber.

URI: The address of the presentity whose presence information has been requested.

Subscriber count: The number of endpoints who have requested information about that particular presentity.

To view the list of all subscribers who are requesting information about a particular presentity, click on the presentity's URI.

Subscribers

Status > Applications > Presence > Subscribers

This page lists each endpoint that has requested information about one or more presentities whose information is managed by (i.e. published to) the local presence server. Endpoints requesting this information are listed here regardless of whether they are registered locally or to a remote server.

Note: FindMe users will not be listed here as a FindMe entity cannot subscribe to presence information. However, one or more of the endpoints that make up a FindMe user may be requesting presence information, in which case that endpoint will be listed here.

URI: The address of the endpoint that has requested presence information.

Subscription count: The number of local presentities about whom this endpoint is requesting information.

To view the list of all local presentities whose information is being requested by a particular endpoint, click on the endpoint's URI.

You are here: [Status](#) > [Applications](#) > [Presence](#) > [Publishers](#)

URI	Document count
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	2
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	2
sip:10.10.10.10@10.10.10.10	2
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	2
sip:10.10.10.10@10.10.10.10	2
sip:10.10.10.10@10.10.10.10	2
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	2
sip:10.10.10.10@10.10.10.10	2

You are here: [Status](#) > [Applications](#) > [Presence](#) > [Presentities](#)

URI	Subscriber count
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1

You are here: [Status](#) > [Applications](#) > [Presence](#) > [Subscribers](#)

URI	Subscription count
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1
sip:10.10.10.10@10.10.10.10	1

Filter

Overview

The OCS Relay application is required in deployments that use both Microsoft Office Communicator (MOC) clients and FindMe, where they both use the same domain. It enables the VCS to:

- share FindMe presence information with MOC clients
- register FindMe users to a Microsoft Office Communications Server (OCS) so that the OCS can forward calls to FindMe aliases



Deployments where the MOC clients and FindMe do not use the same domain do not require use of the OCS Relay application.

Configuring OCS Relay

VCS Configuration

The **OCS Relay** page (**Applications > OCS Relay**) allows you to enable and disable the OCS Relay application on the VCS, and configure the settings it uses. The options are:

OCS Relay mode

This field allows you to enable (**On**) and disable (**Off**) the OCS Relay application on the VCS.

OCS Relay domain

The OCS Relay is used in deployments where MOC clients and FindMe aliases use the same domain. The domain to be used must already be configured on the VCS (**VCS configuration > Protocols > SIP > Domains**). You can then select the domain from the drop-down menu.

OCS Relay routing prefix

To create a connection between the VCS and the OCS, you must have already configured a neighbor zone on the VCS with details of the OCS. In order for the OCS Relay application to be able to route requests to this OCS, you must then:

1. Configure the VCS with an **OCS Relay routing prefix**.
2. Configure a search rule for the OCS neighbor zone that has a pattern match for that **OCS Relay routing prefix**.

This will ensure that all requests with the specified prefix are routed directly to the OCS.

Other requirements

There are a number of other steps required in order to successfully set up a connection between the VCS and OCS. These include:

- creating an appropriate neighbor zone
- configuring Call Policy
- configuring Presence

As this is a complex procedure beyond the scope of this Administrator Guide, you are recommended to refer to the [Microsoft OCS 2007 \(R1 and R2\) and VCS Control Deployment Guide \[24\]](#) which describes in detail all the steps required.

Viewing OCS Relay status

The **OCS Relay status** page (**Status > Applications > OCS Relay**) lists all the FindMe aliases being handled by the OCS Relay application, and shows the current status of each, as follows:

Registration state

This indicates whether the FindMe alias has registered successfully with OCS. This allows OCS to forward calls to the FindMe alias.

Subscription state

This indicates whether the OCS Relay application has subscribed successfully to the FindMe alias's presence information. This will allow MOC clients to view the presence information of FindMe users.

Presence State

This shows the presence information currently available for the FindMe alias.

Overview

What is FindMe?

FindMe is a form of User Policy, which is the set of rules that determines what happens to a call for a particular user or group when it is received by the VCS.

The FindMe feature lets you assign a single FindMe ID to individuals or teams in your enterprise. Users can set up a list of locations such as "at home" or "in the office" and associate their devices with those locations. They can then specify which devices are called when their FindMe ID is dialed, and what happens if those devices are busy or go unanswered. Each user can specify up to 15 devices and 10 locations.

The FindMe feature means that potential callers can be given a single FindMe alias on which they can contact an individual or group in your enterprise — callers won't have to know details of all the devices on which that person or group might be available.

To enable this feature you must purchase and install the FindMe option key.

Standard operation is to use the VCS's own FindMe manager. However, you can use an off-box FindMe manager; this feature is intended for future third-party integration.

Users configure their FindMe settings by logging into their user account.

How are devices specified?

When configuring their user account, users are asked to specify the devices to which calls to their FindMe ID are routed.

It is possible to specify aliases and even other FindMe IDs as one or more of the devices. However, care must be taken in these situations to avoid circular configurations.

For this reason, it is recommended that users specify the physical devices they want to ring when their FindMe ID is called by entering the alias with which that device has registered.

Process overview

When the VCS receives a call for a particular alias it applies its User Policy as follows:

- It first checks to see if FindMe is enabled. If so, it checks if the alias is a FindMe ID, and, if it is, the call is forwarded to the aliases associated with the active location for that user's FindMe configuration.
- If FindMe is not enabled, or the alias is not a FindMe ID, the VCS continues to search for the alias in the usual manner.



User Policy is invoked after any Call Policy configured on the VCS has been applied. See the [Search process](#) section for more information.

Who must do what before FindMe can be used?

The following steps are required for the use of FindMe after the feature has been installed:

1. The VCS administrator [enables and configures FindMe](#).
2. The VCS administrator must define a [Cluster name](#) (even if the VCS is not part of a cluster).
3. The VCS administrator decides whether to use a local or a remote [login account authentication](#) service.
4. The VCS administrator [creates a user account](#) for each user or team of people who require a FindMe ID.
5. If remote authentication is being used, the VCS administrator must also set up [User groups](#).
6. The owner of the FindMe ID [configures their account settings](#).



See the [VCS Deployment Guide - FindMe \[29\]](#) for more details on setting up FindMe accounts.

Recommendations when deploying FindMe

- The FindMe ID should be in the form of a URI, and should be the individual's primary URI.
- Endpoints should not register with an alias that is the same as an existing FindMe ID. You can prevent this by including all FindMe IDs on the Deny List.

Example

Users at Example Corp. have a FindMe ID in the format *john.smith@example.com*. Each of the user's endpoints are registered with a slightly different alias that identifies its physical location. For example their office endpoint is registered with an alias in the format *john.smith.office@example.com* and their home endpoint as *john.smith.home@example.com*. Both of these endpoints are included in the list of devices to ring when the FindMe ID is dialed. The alias *john.smith@example.com* is added to the Deny List, to prevent an individual endpoint registering with that alias.



FindMe is supported by clustering. For information about how FindMe information is managed across peers in a cluster, refer to the [Clustering and FindMe](#) section.

Enabling FindMe on the Cisco VCS

Configuring FindMe

Standard FindMe User Policy uses the VCS's own local FindMe manager, but advanced deployments can connect to a third-party manager.

To configure FindMe:

- **Applications > FindMe > Configuration.**
You are taken to the **FindMe Configuration** page.
- [xConfiguration Policy FindMe](#)

FindMe mode

Determines whether or not FindMe is enabled, and if a third-party manager is to be used.

Off: disables FindMe.

On: enables FindMe and uses the VCS's local FindMe manager.

Third Party Manager: enables FindMe and uses a FindMe manager located on an off-box system. This feature is intended for advanced deployments with third-party integrators.



Call Policy is always applied regardless of the FindMe mode.

The following options apply when **FindMe mode** is **On**:

Caller ID

Determines how the source of an incoming call is presented to the callee.

Incoming ID: displays the address of the endpoint from which the call was placed.

FindMe ID: displays the **FindMe ID** associated with the originating endpoint's address. This means that if the recipient subsequently returns that call, all the devices associated with that FindMe account will be called. For H.323 calls placed through an ISDN gateway, the E.164 **Phone number** associated with the FindMe account is signaled instead as that is a more appropriate number to dial when returning the call. Note that the ISDN gateway must be registered to the same VCS as the call recipient.

Device creation message

The text entered here is displayed to users when they add a device to their FindMe configuration. It can be used to provide information about how to format the device address or number, for example any dial prefixes that must be included.

A limited set of HTML markup is supported in the message which is previewed in the window at the bottom of the page when you click **Save**. The following tags (without any attributes) are allowed:

b i tt big small strike s u em strong cite dfn samp kbd var abbr acronym sub sup ins del br

**** is also supported, but the URL can only contain A-Z 0-9, dot, "?" "=" and "%"; note that the URL is relative to the current page so you must prefix it with, for example, http:// if you want to refer to an external site.

An example message could be:

Phone numbers: use the prefix 9

Endpoints: use the suffix @video.test.com

The following options apply when **FindMe mode** is **Third Party Manager**:

Protocol

The protocol used to connect to the third-party manager.

Address

The IP address or domain name of the third-party manager.

Path

The URL of the third-party manager.

Username

The username used by the VCS to log in and query the third-party manager.

Password

The password used by the VCS to log in and query the third-party manager.

Searching for FindMe accounts

The **User search** page lets you search for user accounts by their related FindMe details such as a FindMe ID or device alias.

This search feature is useful if, for example, you have a device alias but do not know to whom it belongs, or you have a URI and are not sure if it is a FindMe ID or a device alias.

To search for a user account:

- **Applications > FindMe > Search**

Enter the FindMe ID, username, device address or number you want to search for and click **Search**. Note that the search process performs an exact match against the value entered here — "contains" and wildcard searches are not supported.

All matching user accounts are listed. You can review an account's details by clicking **View/Edit**. See [Maintaining user accounts](#) for more information.

Note that if you are part of a large enterprise with, for example, Cisco TMS managing several VCS clusters, the search may find users and devices in other VCS clusters. You can only view (and not edit) the details of accounts that are not managed in your cluster. See [Clustering and FindMe](#) for more information.

Principal devices

A user's account should be configured with one or more principal devices. These are the main devices associated with that account.

Users are not allowed to delete or change the address of their principal devices; they can only change the **Device name** or **Picture**. This is to stop users from unintentionally changing their basic FindMe configuration.

Principal devices are also used by the VCS to decide which FindMe ID to display as a **Caller ID** if the same device address is associated with more than one FindMe ID.

Only an administrator (and not users themselves) can configure which of a user's devices are their principal devices. See [Maintaining user accounts](#).

About FindMe

FindMe lets you control how you are contacted: on any device, at any location, through a single FindMe ID.

You can set up a list of locations such as "at home" or "in the office" and associate your devices (endpoints, Movi, mobile phones and so on) with those locations. You can also set up rules to redirect calls if your devices are busy or unanswered.

For example, you could set up your FindMe so that it calls you on your desktop videophone first. If there's no answer after 10 seconds it diverts the call to your mobile phone, or if your desktop videophone is busy it could divert the call to your colleague's telephone instead.

Your system administrator can also set up a group FindMe for a team of people such as a support desk. This works by calling all the devices associated with that group when the single FindMe group ID is called.

Your administrator may have also configured your external telephone number to route to your FindMe ID.

FindMe user accounts

Each FindMe ID has an associated user account. After the system administrator has set up your account, you can log in to it using a web interface and configure it with details of your work locations and the devices on which you want to be contacted at each location.

After you have set up your device and location details all you typically need to do on an ongoing basis is to indicate your current active location. However, you can change or set up additional devices and locations as often as you want.

Individual versus group FindMe

The difference between individual and group FindMe accounts is what happens if one of the devices in the primary list is busy.

For *individuals*, it is assumed that you can only take calls on one device at a time, therefore if *any* devices in the Primary list are busy, the call immediately diverts to the devices in the Busy list.

For *groups*, it is assumed that more than one person is available to take calls, so the call only diverts immediately to the devices in the Busy list if *all* devices in the Primary list are engaged.

Accessing the FindMe home page

To configure your FindMe user account, log in using a web browser as described below:

The diagram illustrates the process of logging into the FindMe system. It starts with a 'Login' page (Step 1) containing 'User Login' and 'Administrator Login' buttons. An arrow points from 'User Login' to a login form (Step 2) with fields for 'Username' (john.smith) and 'Password' (***), and a 'Login' button. An arrow from the 'Login' button points to the 'TANDBERG FindMe™' home page (Step 3). The home page displays 'My devices' (Office E20, Home T1700, Movi, Mobile) and 'My locations' (Office, Home) with their respective forward and divert settings.

1 Go to the **Login** page using the link provided to you by your system administrator. Click **User login**.

2 Enter the **Username** and **Password** provided to you by your System Administrator. Click **Login**.

3 You are taken to the **FindMe home page**. From here you can [configure your FindMe user account](#) and set your current active location.

Configuring your FindMe user account

FindMe home page

This page summarizes your FindMe account. It shows your configured devices and locations, and your currently **Active** location.

From here you can change your **Active** location or use the **Add**, **Edit** and **Delete** buttons to modify your device and location details.

FindMe account details

The FindMe ID and external telephone number of the account being configured.

My devices

A list of all the devices configured for your FindMe account.

Click **Edit** next to the device whose details you want to change, to open a new window where you can modify the device name, address (alias or number) or picture.

Click **Delete** to remove a device from the list.

Click **Add new device** to open a new window where you can set up a new device. You can add up to 15 devices.



When your account was set up, your system administrator may have configured one or more principal devices. These are the main devices associated with your account. You cannot delete or change the address of your principal devices; you can only change the device name or picture. Contact your administrator if you need to change your principal devices.



Ensure that none of the active location's primary devices are set to Autoanswer.

If they are, the system will consider the call to have been answered when Autoanswer is initiated, and so it will not divert the call to any other devices.

Active location

Shows the currently **Active** location. When someone calls your FindMe ID the devices associated with your active location are called.

To change your active location, click on the radio button next to the required location name. One of your FindMe locations must always be active.



If no devices or locations are configured, all calls to your FindMe will be rejected.

Change Password

Click here to change the password used to access your FindMe account (if your account name and password are managed locally within the VCS). This goes to a new page where you can enter a new password. Note that passwords are case sensitive.

Logout

Click here to exit the FindMe home page.

My locations

Lists the locations against which you can associate different combinations of your devices.

Forward calls to shows the primary devices that are called when your FindMe ID is first dialed (where that is the currently active location). If more than one device is listed here, they will all ring at the same time.

Divert when busy to shows the devices that are called if the primary devices are busy.

Divert when not answered to shows the devices that are called if the primary devices are not answered after a specified time.

Click **Edit** next to the location whose details you want to change, to open a new window where you can select the primary devices associated with that location and the devices to divert to when the primary devices are busy or not answered.

Click **Delete** to remove a location from the list. Note that you cannot delete the active location.

Click **Add new FindMe location** to open a new window where you can set up a new location and its associated devices. You can add up to 10 locations.

Configuring your FindMe user account

Defining device details

This page is used to define a new device such as an endpoint, Movi client or telephone, or to change the details of an existing device.

To get here from the FindMe home page, click **Add new device** to define a new device, or click **Edit** against the device whose details you want to modify.

Device name

Enter a description of how you refer to the device. This name is not seen by the people calling that device.

Address or number

The contact address / alias / number of the device. Depending on the type of device you should use the following formats:

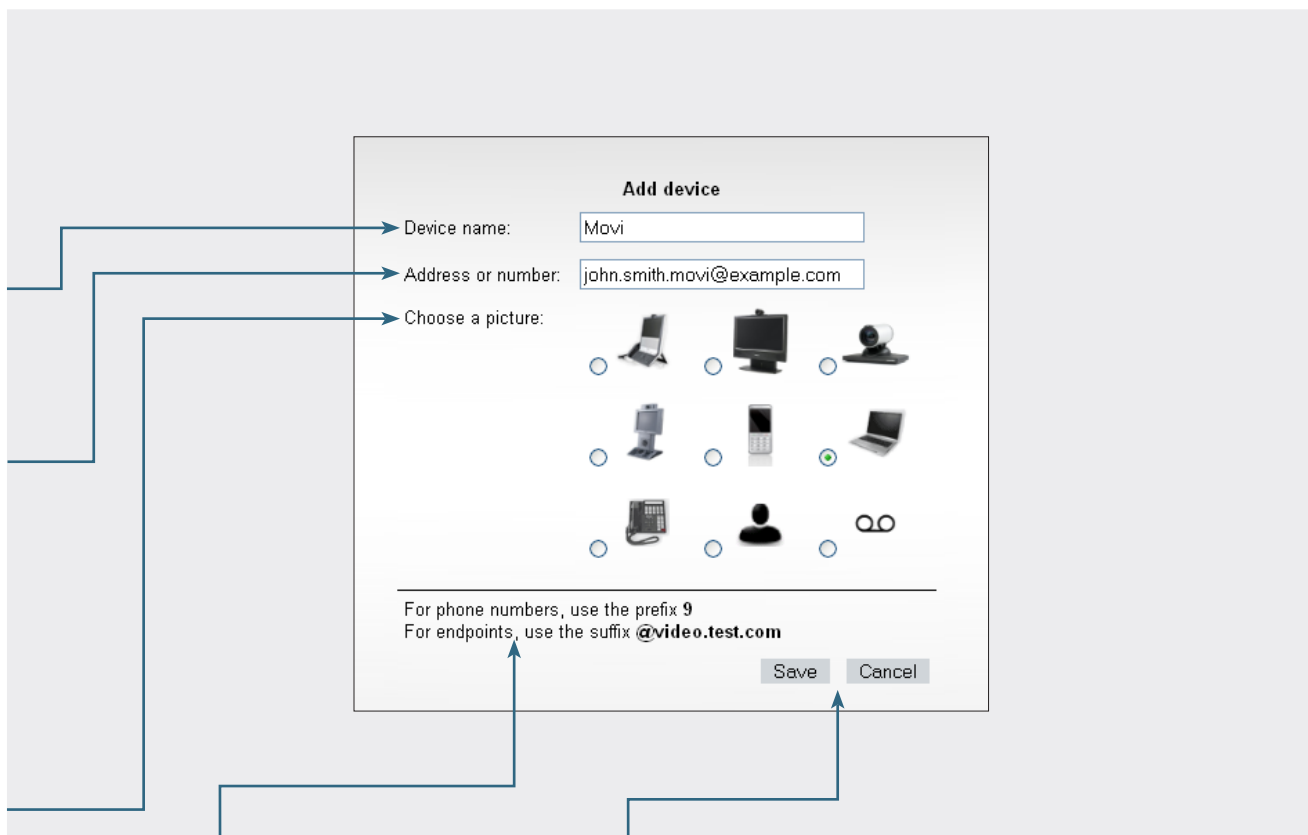
- video endpoints: enter any alias with which the device is registered
- 3G mobile phones:
 - to route video to your mobile phone, you must have a 3G gateway - enter the gateway's prefix followed by the mobile phone number
 - to route voice only, enter the mobile phone number along with any prefixes required by your dial plan for external calls
- telephones: enter the extension number (for internal calls) or the full telephone number, along with any necessary prefixes

Picture

Choose a picture to associate with the type of device.

Note that the pictures only act as a guide and are for your own benefit. They have no effect on how calls to that device are handled.

For example you could use  to indicate a forwarding address of a colleague, or  for voicemail.



Add device

Device name:

Address or number:

Choose a picture:

For phone numbers, use the prefix 9
For endpoints, use the suffix @video.test.com

Information

The system administrator can include instructions here. It could, for example, show you how to format your device addresses.

Save / Cancel

Click **Save** to confirm your changes and return to the FindMe home page.
Click **Cancel** to discard your changes and return to the FindMe home page.

Configuring your FindMe user account

Defining location details

This page is used to define the details of a new location, or to change the devices associated with an existing location.

To get here from the FindMe home page, click **Add new FindMe location** to define a new location or click **Edit** against the location whose details you want to modify.

Location name

Enter a description that describes a place or mode of working such as "Office" or "Home". This name is not seen by the people that call you.

Primary devices

Select the devices to call when your FindMe ID is first dialed (and where this is your currently active location). You can choose from all of the devices you have set up in your **My devices** list.

If more than one device is selected, they will all ring at the same time.

more... / less...

Click **more...** to expand the screen and specify how the location should work if your primary devices are busy or unanswered.

Click **less...** to hide this information.

 Busy devices

For an **individual** FindMe account the busy devices ring immediately if any of the primary devices for that location are busy.

For a **group of people** FindMe account the busy devices only ring immediately if *all* of the primary devices are busy. (If *some* of the primary devices are busy, the rest will ring for the specified time before the call diverts to the busy devices.)

If the active location has no "divert when busy to" devices, the caller will get a busy response if any of the primary devices are busy (or if all devices are busy if it is a **group** FindMe).

If the primary devices are busy

Select the devices to call if any of your primary devices are busy.

If the primary devices are not answered

Select the amount of time in seconds you want the devices in the primary list to ring, and then select the devices to call if there is no answer after that time. If no secondary devices are selected, the call will terminate after the chosen duration, and the caller will receive a "no answer" response.

Alternatively, you can select an option that the devices will ring **until caller hangs up**.

Save / Cancel

Click **Save** to confirm your changes and return to the FindMe home page.

Click **Cancel** to discard your changes and return to the FindMe home page

Overview

The VCS's provisioning server provides basic device provisioning for registered Movi users, without the need for TMS.

The **Starter Pack** option key must be installed to use basic device provisioning. It cannot be used in combination with Device Provisioning through the TMS Agent and TMS Provisioning Directory. Note that the **Starter Pack** option key is designed for single box deployments and is only available as a pre-configured factory setting.

Starter Pack

The Starter Pack is suitable for small enterprises. It provides basic VCS functionality and has limited licenses.

The following features are included with the Starter Pack:

- [Expressway](#)
- [FindMe](#)

The following license restrictions apply:

- 50 registrations
- 5 calls (any combination of traversal and non-traversal calls)

Note that installing additional call license option keys will have no effect while the **Starter Pack** option key is present.

Device authentication

The provisioning server supports [device authentication](#). Provisioned devices' credentials can be authenticated against the VCS's local authentication database (or the LDAP directory if remote authentication is selected) when they attempt a provisioning request and when they register with the VCS.

Phone book support

Provisioned devices are served a phone book directory of user accounts.

Configuration

The provisioning server is configured using the **Provisioning** page (**Applications > Provisioning**). The server is automatically enabled when the **Starter Pack** option key is installed.

Device bandwidth limits

Enabling the **Provision bandwidth limits** option allows you to configure the maximum incoming and outgoing bandwidth (in kbps) values to set on the provisioned devices.



You can monitor the provisioning server on the [Provisioning status](#) page.

Provisioning users

To provision individual users, you must set up user accounts.

When you configure a user account, a provisioned device URI is assigned to that user and enables provisioning for that device by default. User accounts are also used to configure a user's FindMe settings. See [Maintaining user accounts](#) for more information.



See the [VCS Deployment Guide - VCS Expressway Starter Pack \[34\]](#) for full details on setting up Starter Pack provisioning.

Overview

TMS Agent is a process that runs on the VCS to manage FindMe and Device Provisioning data.

It acts on behalf of Cisco TMS so that Cisco TMS is not a single point of failure, and enables each VCS to share the load. It supports the replication of FindMe and provisioning data, sharing the data among cluster peers as well as the central Cisco TMS, providing resilience in case of connection failures between any VCS and Cisco TMS.

TMS Agent is installed as part of the VCS Platform and requires no configuration on the VCS, other than ensuring the default password is changed (see **TMS Agent account passwords** below).

- You must use Cisco TMS to create and manage Device Provisioning data.
- FindMe accounts may be set up using Cisco TMS or VCS.

FindMe™

The TMS Agent replicates changes to FindMe account information across peers in a VCS cluster (FindMe account changes can be made on any peer, not just the master), across to Cisco TMS and also across to other VCS clusters managed by the same Cisco TMS.

Note that the FindMe option key must be installed on the VCS.

Device Provisioning

The TMS Agent works with the TMS Provisioning Directory to replicate and distribute the provisioning information and phonebook from Cisco TMS via VCSs to endpoint devices. VCSs cache and replicate data among themselves in case connection to Cisco TMS is lost.

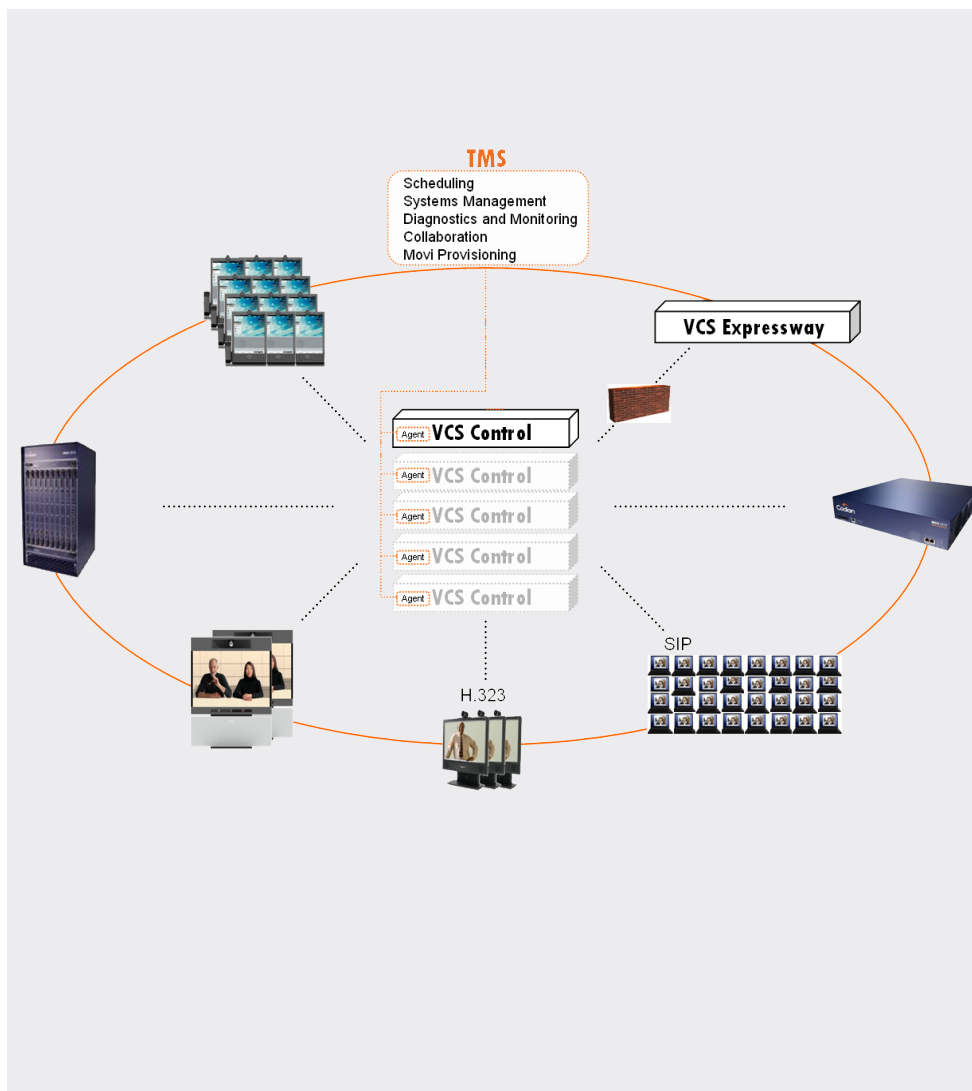
Note that the Device Provisioning option key must be installed on the VCS.

TMS Agent account passwords

TMS agent is accessed via two accounts: one for connecting via LDAP into the TMS Agent database, and one for managing the replication of the TMS Agent database. These accounts are only used by the internal processes running on the VCS and Cisco TMS. System administrators must not use these accounts.

These accounts have a username of cn=Directory Manager and a default password of TANDBERG (all upper case). Both passwords must be changed as soon as possible to maintain security of the VCS data. Warnings are shown on the web UI and the CLI if either account has the default password set.

- If your VCS uses Cisco TMS as an external manager, you must use Cisco TMS to change the passwords on these accounts. Instructions for this are in the [Clustering deployment guide \[27\]](#).
- If your VCS is not managed by Cisco TMS, you have to change these passwords by logging into the VCS as the root user. Instructions for this are contained in the online help; follow the link associated with the VCS's TMS Agent password warning message. If your VCS is subsequently reconfigured to use TMS, the password must first be reset to the default value of TANDBERG.



Maintenance

This section describes the pages that appear under the **Maintenance** menu of the Cisco VCS web interface.

These pages allow you to perform the following tasks:

- [upgrade to a new release of software](#)
- [downgrade to a previous version of software](#)
- [install and delete option keys](#)
- [manage security certificates](#)
- [enable advanced account security](#)
- [manage administrator and FindMe user accounts and passwords](#)
- [create and restore backups](#)
- [create a system snapshot](#)
- [view incidents and configure incident reporting](#)
- [use built-in tools to check patterns and locate aliases](#)
- [view a list of all ports used by the VCS](#)
- [restart, reboot or shut down the VCS](#)

This section also provides information on:

- [restoring the system to its default settings](#)
- [password encryption](#).



Overview

You can install new releases of the VCS software components on your existing hardware. Component upgrades can be performed in one of two ways:

- [using the web interface \(HTTP/HTTPS\)](#) - this is the recommended process
- [using secure copy \(SCP/PSCP\)](#)

This section describes how both of these methods are used to perform upgrades.



To avoid any performance degradation you are recommended to upgrade VCS components while the system is inactive.



You can also upgrade the **VCS platform** component using Cisco TMS. See the Cisco TMS documentation for more information.



For specific information about upgrading peers in a cluster, refer to the [Clustering deployment guide \[27\]](#).

VCS software components

All existing installed components are listed on the **Upgrade** page (**Maintenance > Upgrade**), showing their current version and associated release key where appropriate.

The main component is the **VCS platform**. Other components typically include **OCS Relay**, the **VCS Database** and **Device Provisioning**.

An upgrade to the **VCS platform** component will typically include automatic upgrades of some or all of the other components. However, you can independently upgrade the other components if required to do so. The upgrade process ensures that compatibility is maintained across all components.

Prerequisites

The upgrade requires you to have:

- a valid **Release key**, if you are upgrading to the next major release of the **VCS platform**, for example from X4.1 to X5.0; it is not required for dot releases, for example X5.0 to X5.1
- a software image file for the component you want to update
- release notes for the software version you are upgrading to — additional manual steps may be required

Contact your Cisco representative for more information on how to obtain these.

Backing up before upgrading

You should backup your system configuration before upgrading. Click **System backup** to go to the [Backup and restore](#) page.



If you later need to downgrade to an X4 (or earlier) release you will have to restore a backup made against that previous release.

Upgrading and option keys

All existing option keys are retained through the upgrade from one version of the VCS platform to the next, including upgrades to the next major release. However, you are recommended to take note of your existing option keys before performing the upgrade.

New features may also become available with each major release of the VCS platform component, and you may need to install new option keys to take advantage of these new features. Contact your Cisco representative for more information on all the options available for the latest release of VCS software.

Installing and rebooting

Upgrading the **VCS platform** component is a two-stage process. First, the new software image is uploaded onto the VCS. At the same time, the current configuration of the system is recorded, so that this can be restored after the upgrade. During this initial stage the system will continue running on its existing software version, and all normal system processes will continue.

The second part of the upgrade involves rebooting the system. It is only during the reboot that the VCS installs the new software version and restores the previous configuration.

This means that you can upload the new software to your system at any time, and then wait until a convenient moment (for example, when no calls are taking place) to install the new version by rebooting the system.



Any configuration changes made between the software upload and the reboot will be lost when the system restarts using the new software version.

The upgrade of components other than the **VCS platform** does not involve a system reboot, however the services provided by that component will be temporarily stopped while the upgrade process completes.

Upgrade procedure

Before starting the upgrade, ensure that you have:

- the software image file for the component you want to upgrade, and it is stored in a network location that is locally accessible from your client computer
- a valid 16-character **Release key**, if you are upgrading to a new major release of the VCS platform (it is not required if you are upgrading to a dot release, for example from X4.0 to X4.1)

Upgrading using the web interface

To upgrade a software component using the web interface:

1. Go to the **Upgrade** page (**Maintenance > Upgrade**).
 2. Click **Browse** and select the software image file for the component you want to upgrade.
 3. Enter the **Release key** if required. If you have cut and pasted the release key, ensure there are no leading or trailing spaces.
 4. Click **Upgrade**. The VCS detects which component you are upgrading based upon the selected software image file.
 5. For upgrades to the **VCS platform** component, the **Upgrade confirmation** page is displayed:
 - a. Check the expected **New software version** number is displayed and click **Continue with upgrade**.
The **System upgrade** page opens and displays a progress bar while the software installs.
When the install is complete, you need to reboot the VCS for the new version to take effect. Rebooting causes all current calls to terminate, and all current registrations to be ended. This page indicates the number of active calls and registrations on your VCS so that you can reboot it at an appropriate time.
 If you do not reboot the system immediately, you should refresh this page before restarting to check the current status of calls and registrations.
 - b. Click **Reboot system**. After the reboot is complete you are taken to the **Login** page.
- For upgrades to other components, the software is automatically installed. No reboot is required.

The upgrade is now complete. The **Overview** and **Upgrade** pages now show the upgraded software component version numbers.

Upgrading using secure copy (SCP/PSCP)

To upgrade using a secure copy program such as SCP or PSCP (part of the PuTTY free Telnet/SSH package) you need to transfer two files to the VCS:

- A text file containing just the 16-character Release Key (required for the **VCS platform** component only). Ensure there is no extraneous white space in this file.
- The file containing the software image.

To transfer these files:

1. If you are upgrading the **VCS platform** component, upload the Release Key file using SCP/PSCP to the **/tmp/** folder on the system. The target name must be **release-key**, for example:

```
scp release-key root@10.0.0.1:/tmp/release-key
```

Enter the **root** password when prompted.
The Release Key file must be uploaded before the image file.
2. Upload the software image using SCP/PSCP.

For the **VCS platform** component:

- Upload to the **/tmp** folder on the system. The target name must be **/tmp/tandberg-image.tar.gz**, for example:

```
scp s42700x5.tar.gz root@10.0.0.1:/tmp/tandberg-image.tar.gz
```

For other components:

- Upload to the **/tmp/pkgs/new/** folder on the system, preserving the file name and extension, for example:

```
scp root@10.0.0.1:/tmp/pkgs/new/ocs-relay.tlp
```

3. Enter the **root** password when prompted.

The software installation begins automatically. Wait until the software has installed completely. This should not take more than five minutes.

4. If you have upgraded the **VCS platform** component, log in to the VCS, either using the web interface or CLI, and [reboot the VCS](#). After about five minutes the system will be ready to use.



If you make any further configuration changes before rebooting, those changes will be lost when the system restarts, so you are recommended to reboot your system immediately.

Downgrade procedure



If you need to downgrade to an X4 (or earlier) release of the **VCS Platform**, configuration changes, including changes made to FindMe or Provisioning, will be lost.

- When the downgrade has completed you will have to restore a backup of the system configuration that was made against the release you have just reinstalled.
- Other manual steps may be required — you must review the release notes for the version you are downgrading from.

Reinstalling a previous software version

The procedure for downgrading to an earlier software version is the same as when [upgrading software](#). You need to have:

- a valid release key
- a software image file (of the software version you are reverting back to)

You should already have obtained these when the previous version of the software was installed on your VCS.

Backing up current configuration

You should backup your system configuration before downgrading. Click **System backup** to go to the [Backup and restore](#) page.

Overview

Your VCS may have been shipped with one or more optional features pre-installed. Additional VCS features can be added to your existing system by the installation of option keys.

To view the list of options installed currently, go to **Maintenance > Option keys**. You will be taken to the **Option keys** page.

The options that you may see here include:

- **Expressway**: enables the VCS to work as an Expressway™ firewall traversal server.
- **H.323 to SIP Interworking gateway**: enables H.323 calls to be translated to SIP and vice versa.
- **FindMe™**: enables [FindMe](#) functionality.
- **Dual Network Interfaces**: enables the LAN 2 port on your VCS Expressway.
- **Device Provisioning**: allows VCS to provision endpoints with configuration information on request and to supply endpoints with phone book information. (Endpoints including Movi v2.0 or later, and E20 v2.1 or later can request to be provisioned.) Note that the VCS must use Cisco TMS as its external manager to obtain configuration and phone book information for distribution.
- **Starter Pack**: allows the VCS to offer basic device provisioning for registered Movi users, without the need for Cisco TMS (see [Provisioning \(Starter Pack\)](#)).
- **Traversal calls**: determines the number of traversal calls allowed on the VCS at any one time. Note that traversal calls that are passing through the VCS from one neighbor to another but where neither endpoint in the call is locally registered will still be counted as one traversal call. See the [What are traversal calls?](#) section for more information.
- **Non-traversal calls**: determines the number of non-traversal calls allowed on the VCS at any one time. Note that non-traversal calls that are passing through the VCS from one neighbor to another but where neither endpoint in the call is locally registered may or may not require a non-traversal call license, depending on the [Call routed mode](#) setting. Note that a non-traversal call on a VCS Expressway will consume a traversal license if there are no non-traversal call licenses available.

- **Registrations**: the number of concurrent **registrations** allowed on the VCS. An endpoint can register with more than one alias and this will be considered to be a single registration. However, an endpoint that supports both SIP and H.323 and registers using both protocols will count as two registrations. H.323 systems such as gateways, MCUs and Content Servers can also register with a VCS, and these will each count as one registration.
- **TURN Relays**: the number of concurrent TURN relays that can be allocated by this VCS. See the [TURN services](#) section for more information.
- **Encryption**: indicates that AES encryption is supported by this software build.
- **Advanced account security**: enables advanced security features and restrictions for high-security installations.

Contact your Cisco representative for more information on how to purchase these features.

After the appropriate option key has been purchased, it must be installed. You can do this through the web interface or through the CLI.

Adding option keys using the web interface

To add options using the web interface:

- **Maintenance > Option keys**.

You are taken to the **Option keys** page.

The first section on this page lists the keys that are already installed on your system along with a description of the options they provide.

The **System information** section tells you about the hardware and options that currently make up your system.

1. In the **Add option key** field, enter the 20-character key that has been provided to you for the option you wish to add.
2. Click **Add option**.



Some option keys require that the VCS is restarted before the option key will take effect. In such cases you will receive a warning on the web interface, which will remain in place as a reminder until the system has been restarted. However, you can continue to use and configure the VCS in the meantime.

Adding option keys using the CLI

To return the indexes of all the option keys that are already installed on your system:

- [xStatus Options](#)

To add a new option key to your system:

- [xConfiguration Option \[1..64\] Key](#)



When using the CLI to add an extra option key, you can use any unused option index. If you chose an existing option index, that option will be overwritten and the extra functionality provided by that option key will no longer exist. To see which indexes are currently in use, type **xConfiguration option**.

Overview

For extra security, you may want to have the VCS communicate with other systems (such as LDAP servers, neighbor VCSs, or clients such as SIP endpoints) using TLS encryption.

For this to work successfully in a connection between a client and server:

- The server must have a certificate installed that verifies its identity. This certificate must be signed by a Certificate Authority (CA).
- The client must trust the CA that signed the certificate used by the server.

The VCS allows you to install appropriate files so that it can act as either a client or a server in connections using TLS. The VCS can also authenticate client connections (typically from a web browser) over HTTPS. You can also upload certificate revocation lists (CRLs) for the CAs used to verify LDAP server and HTTPS client certificates.



For an endpoint to VCS connection, the VCS acts as the TLS server. For a VCS to LDAP server connection, the VCS is a client. For a VCS to VCS connection either VCS may be the client with the other VCS being the TLS server. For HTTPS connections the web browser is the client and the VCS is the server.



TLS can be difficult to configure. For example, when using it with an LDAP server it is recommended that you confirm that your system is working correctly before you attempt to secure the connection with TLS. It is also recommended that you use a third party LDAP browser to verify that your LDAP server is correctly configured to use TLS.



Be careful not to allow your CA certificates or CRLs to expire as this may cause certificates signed by those CAs to be rejected.

For more information on setting up security certificates, refer to the [Certificate Creation and Use Deployment Guide \[32\]](#).

Enabling security

To enable certificate security using the web interface:

- **Maintenance > Security certificates.**

You are taken to the **Security certificates** page.



Certificate and certificate revocation list (CRL) files can only be loaded via the web interface. They cannot be installed using the CLI.

Trusted CA certificate

This section manages the list of certificates for the Certificate Authorities trusted by this VCS. Certificates presented to the VCS must be signed by a trusted CA on this list and there must be a full chain of trust to the root CA.

- To upload a new file of CA certificates, **Browse** to the required PEM file and click **Upload CA certificate**. This will replace any previously uploaded CA certificates.



If certificate revocation list checking for TLS encrypted [LDAP server connections](#) (for account authentication) is enabled, the necessary PEM encoded CRL data must be included within the trusted CA certificate file.

- Click **Reset to default CA certificate** to replace the currently uploaded file with a default list of trusted CA certificates.
- Click **Show CA certificate** to view the currently uploaded file.

Server certificate data

This section is used to upload the VCS's server certificate. This certificate is used to identify the VCS when it communicates with client systems using TLS encryption, and with web browsers over HTTPS.

- Use the **Browse** buttons to select the **server certificate** PEM file and the **server private key** PEM file that is used to encrypt it. After selecting both files, click **Upload server certificate data**. The private key must not be password protected.
- Click **Reset to default server certificate** to replace the current server certificate with the VCS's default certificate.
- Click **Show server certificate** to view the currently uploaded server certificate file.

HTTPS client certificate validation

The **Client certificate validation** setting controls whether client systems (typically web browsers) that communicate with the VCS over HTTPS have to present a valid client certificate before the connection can be established. Note that a restart is required for changes to this setting to take effect.



If you enable client certificate validation your browser will be able to use the VCS web interface only if it has a valid client certificate that is signed by a CA in the VCS's trusted CA certificate list. Ensure your browser (the client system) has a valid (in date and not revoked by a CRL) client certificate before enabling this feature. You can test if a client certificate is valid by using the **client certificate test** feature described below.

The procedure for uploading a certificate to your browser may vary depending on the browser type and you may need to restart your browser for the certificate to take effect.

Client certificate revocation list (CRL) file

You are recommended to upload CRL data for the CAs that sign the HTTPS client certificates. Note that CRL checking is applied for every CA in the chain of trust.

- To upload a PEM encoded CRL file, **Browse** to the required file and click **Upload CRL for client certificates**. This will replace any previously uploaded CRL file.
- Click **Remove revocation list** if you want to remove all HTTPS client CRL information from the VCS.



CRL data uploaded here only applies to HTTPS client certificate validation; CRL data intended for validating TLS connections with an LDAP server must be contained within the trusted CA certificate file.

Client certificate test

To verify if a client certificate will be accepted before enabling client certificate validation:

- Click **Browse** to select the required PEM file and then click **Test client certificate file**. The selected file will be checked against the VCS's trusted CA list and the client certificate revocation list. A success or failure message will be displayed.

Overview

The VCS's **Advanced account security** mode is used to configure the VCS for use in highly secure environments.

Enabling advanced account security limits login access to remotely authenticated users using the web interface only, and also restricts access to some VCS features. To indicate that the VCS is in secure mode, a **Classification banner** message is displayed on every web page.



This functionality can only be enabled if the **Advanced account security** option key is installed.

Prerequisites

Before advanced account security mode can be enabled, the VCS must be configured to use [remote account authentication](#) for administrator accounts.



Ensure that the remote directory service is working properly, as after advanced account security is enabled you will not be able to log in to the VCS via the local **admin** account or as root.

You are also recommended to configure your system so that:

- SNMP is disabled
- the session time out period is set to a non-zero value
- HTTPS client certificate validation is enabled
- login account LDAP server configuration uses TLS encryption and has **Certificate revocation list (CRL) checking** set to **All**
- remote logging is disabled
- incident reporting is disabled
- any connection to an external manager uses HTTPS and has certificate checking enabled

VCS functionality: changes and limitations

When in secure mode, the following changes and limitations to standard VCS functionality apply:

- access over SSH, Telnet, and through the serial port is disabled and cannot be turned on
- access over HTTPS is enabled and cannot be turned off
- the command line interface (CLI) is unavailable
- the root account, the admin account and any other local administrator accounts are disabled
- if there are three consecutive failed attempts to log in (by the same or different users), login access to the VCS is blocked for 60 seconds
- immediately after logging in, the current user is shown statistics of when they previously logged in and details of any failed attempts to log in using that account
- administrator accounts with Read-Only or Read-Write access levels cannot view the **Event Log** and **Configuration Log** pages (these pages can only be viewed by accounts with Auditor access level)
- the **Upgrade** page only displays the VCS platform component
- downgrades to version X5.0 or below are not allowed
- the classification banner is displayed on every web page

Enabling advanced account security

To enable advanced account security using the web interface:

- **Maintenance > Advanced account security.**
You are taken to the **Advanced account security** page.

To configure advanced account security using the CLI:

- [xConfiguration Certification AdvancedAccountSecurity Mode](#)

Advanced account security mode

The options for this setting are:

On: puts the VCS into secure mode.

Off: takes the VCS out of secure mode. Note that the Event Log, Configuration Log, call history, search history and registration history are cleared whenever the VCS is taken out of secure mode.



Before advanced account security can be enabled the system checks that all prerequisites are in place. Warnings are also raised for any non-recommended configuration settings.



A system reboot is required for changes to the **Advanced account security mode** to take effect.

Classification banner

Text entered here is displayed as a banner on every page when the VCS is in secure mode.

Accounts overview

About Cisco VCS login accounts

The VCS has two types of login account for normal operation:

- **Administrator accounts:** used to configure the VCS.
- **User accounts:** used by individuals in an enterprise to configure their FindMe profile. They can also be used to enable basic device provisioning for registered Movi users when the **Starter Pack** option key is installed.

Account authentication

Administrator and user accounts must be authenticated before access is allowed to the VCS.

The VCS can authenticate accounts either locally or against a remote directory service, such as Windows Active Directory, using LDAP. The remote option allows administration groups to be set up in the directory service for all VCSs in an enterprise, removing the need to have separate accounts on each VCS.

If a remote source is used for either administrator or user account authentication you also need to configure the VCS with:

- appropriate LDAP server connection settings (see [Account authentication using LDAP](#))
- administrator groups and/or user groups that match the corresponding group names already set up in the remote directory service to manage administrator and user access to this VCS (see [Administrator and user groups](#))

Administrator accounts

Administrator accounts are used to configure the VCS. The VCS has a default **admin** administrator account with full read-write access and can be used to log in to the VCS using the web interface or the CLI.

You can add additional administrator accounts which can only be used to log in through the web interface.

The default **admin** account is managed locally and is always accessible, even if remote administrator account authentication is selected.

See the [Administrator accounts](#) section for more information.

User accounts

User accounts are used by individuals in an enterprise to configure the devices and locations on which they can be contacted through their FindMe ID.

Each user account is accessed using a username and password.

- If local user account authentication is selected, each user account must be created locally by a VCS administrator.
- If remote user account authentication is selected, a VCS administrator must set up user groups to match the corresponding group names in the remote directory service.

Note that if remote user account authentication is selected, only the username and password details are managed remotely. All other properties of the user account, such as the FindMe ID, devices and locations are stored in the local VCS database.

See the [Maintaining user accounts](#) section for more information about defining user account details and their associated FindMe devices and locations, and for enabling basic **Starter Pack** provisioning.



Use Cisco TMS if you need to provision a large number of user accounts.



See the [VCS Deployment Guide - FindMe \[29\]](#) for more details on configuring FindMe and user accounts.

Root accounts

The VCS provides a root account which can be used to log in to the VCS operating system. The **root** account should not be used in normal operation, and in particular system configuration should not be conducted using this account. Use the **admin** account instead.

See the [Root account](#) section for more information.



Remember to change the passwords for the **admin** and **root** accounts from their default values.

Account authentication configuration

To specify where administrator and user accounts are authenticated before access is allowed to the VCS:

- **Maintenance > Login accounts > Configuration.**
You are taken to the **Login account authentication configuration** page.

To specify authentication sources using the CLI:

- [xConfiguration Login Administrator Source](#)
- [xConfiguration Login User Source](#)

Administrator authentication source

Defines where administrator login credentials are authenticated.

User authentication source:

Defines where user login credentials are authenticated.

The authentication source options are:

Remote: credentials are verified against an external credentials directory, for example Windows Active Directory.

Local: credentials are verified against a local database stored on the VCS.

After specifying where accounts are authenticated you must set up the appropriate account details or directory service group details. The **Related tasks** section at the bottom of the page provides links to the relevant pages.



See the [VCS Deployment Guide - Authenticating VCS accounts using LDAP \[30\]](#) for more details on configuring a remote directory service.

Administrator accounts

Default administrator account

The VCS has a default administrator account with full read-write access. This account is used to log in to the VCS using the web interface or the CLI. The username for this account is **admin** (all lower case) and the default password is **TANDBERG** (all upper case).

You cannot delete the default administrator account or change its **admin** username, but you should change the password as soon as possible. Choose a strong password, particularly if administration over IP is enabled.



The default **admin** account is managed locally and is always accessible, even if remote administrator account authentication is selected.



If you forget the password for the **admin** account, you can still log in as another administrator user with **read-write** access and change the password for the **admin** account.

If you do not have any other such administrator users set up, or you have forgotten those passwords as well, it is possible to reset the password for the **admin** account as long as you have physical access to the VCS. See the section [Resetting passwords](#) for details.

Additional administrator accounts

You can add up to 15 additional local administrator accounts. These can be used to log in using the web interface only.



The [Configuration Log](#) records all login attempts and configuration changes made using the web interface, and can be used as an audit trail. This is particularly useful when you have multiple administrator accounts.



It is possible to have more than one administrator session running at the same time. These sessions could be using the web interface, command line interface, or a mixture of both. This may cause confusion if each administrator session attempts to modify the same configuration settings - changes made in one session will overwrite changes made in another session.

Administrator password security

The **Password security** page (**Maintenance > Login accounts > Password security**) lets you determine whether or not administrator passwords and the root password must meet a minimum level of complexity before they are accepted.

If **Enforce strict passwords** is set to **On**, all subsequently configured administrator passwords and root passwords must contain at least 15 ASCII characters made up of at least:

- 2 lowercase letters ['a'..'z']
- 2 uppercase letters ['A'..'Z']
- 2 numeric values ['0'..'9']
- 2 special characters [e.g. '@', '\$']

If you change **Enforce strict passwords** from **Off** to **On**, you will receive a warning if any existing administrator accounts or the root account have passwords that do not meet the security requirements.

If **Enforce strict passwords** is set to **Off**, no checks are made on administrator passwords.



The **Enforce strict passwords** setting affects administrator passwords and the password for the [root account](#) only. It does not affect any other passwords used on the VCS such as in the local authentication database, LDAP server, outbound connection credentials or user account passwords.

- You cannot set a blank password for any administrator account.
- All passwords and usernames are case sensitive.

Maintaining administrator accounts

The **Administrator accounts** page lists all the administrator accounts that have been configured on the VCS, and lets you add, edit and delete accounts.

To go to the **Administrator accounts** page:

- **Maintenance > Login accounts > Administrator accounts.**

Click on the account you want to configure (or click **New** to create a new account, or click **Delete** to remove an account).



Only the **admin** account can be configured if remote administrator account authentication is enabled.

To configure administrator accounts using the CLI:

- [xConfiguration SystemUnit AdminAccount](#)

Name

The username for the administrator account. (Note that some names such as "root" are reserved.)

Password

Enter the password that this administrator will use to log in to the VCS. The password can be up to 16 characters.

All passwords on the VCS are encrypted, so you only see placeholder characters here.

Confirm password

Retype the password entered above.

Account access

Determines the rights for this account. The options are:

Read Write: allows all configuration to be viewed and changed. This provides the same rights as the default **admin** account.

Read Only: allows status and configuration information to be viewed only and not changed. Some pages, such as the Upgrade page, are blocked to read-only accounts.

Auditor: allows access to the **Event Log**, **Configuration Log** and the **Overview** page only.

Account Disabled: web login access to the VCS is not allowed.

Maintaining user accounts

The **User accounts** page lists all the user accounts held in the VCS's local database and lets you add, edit and delete accounts.



If you are part of a large enterprise with, for example, Cisco TMS managing several VCS clusters, the database may contain details of users and devices in other VCS clusters. Different clusters are distinguished by their **Cluster name**. You cannot modify the details of accounts that are not managed in your cluster.

To go to the **User accounts** page:

- **Maintenance > Login accounts > User accounts.**

Click on the account you want to configure (or click **New** to create a new account, or click **Delete** to remove an account).

Creating user accounts

Account details are divided into three sections: general user details, FindMe details and provisioning details (the provisioning section only appears if the **Starter Pack** option key is installed).

Username

The account name. It is used (along with a password) by the user to log in to the VCS and configure their FindMe details. The username cannot be changed after the account has been created.



If remote authentication is enabled, the username defined in the VCS must match the username set up in the remote directory service.

Display name

A free-form display name for the user (rather than the user's **Username** or **FindMe ID**). It is used in phone books and as the caller display name in SIP calls.

Phone number

The numeric caller ID presented when making an outbound call through an ISDN gateway.

To allow call return, this number could be configured to route to the associated FindMe ID by mapping incoming numbers to the FindMe ID using ENUM, search rules or CPL. See the [VCS Deployment Guide - FindMe \[29\]](#) for more information.

Password

The password to be used, along with the **Username**, when logging into this account. Enter the **Initial password** and then enter it again in the **Confirm password** field. Users can change their password after they have logged in.

Note that passwords are case sensitive, and that the password fields are not shown if **Remote** authentication is enabled.

FindMe ID

The FindMe alias — the dialable address — by which the user can be contacted.

The FindMe ID can be any string of up to 60 characters. However, not all endpoints are able to dial aliases with spaces or other non-alphanumeric characters so it is recommended that these are not used in your FindMe IDs.

Principal device address

The address or alias of the user's first principal device. An administrator (or users themselves) can add more endpoint addresses after the account has been created.

Note that if the **Starter Pack** option key is installed, you can choose whether the address of the user's principal device should be set to the default device URI (which follows the same format as the **Provisioned device URI**) or whether to enter an alternative address.

FindMe type

Specifies if the account is for an individual person or a group of people, and affects how calls are diverted when endpoints in the user's primary list are busy.

Individual: calls immediately divert if any primary endpoint is busy.

Group of people: calls immediately divert only if all primary endpoints are busy.

If the **Starter Pack** option key is installed all FindMe IDs are created as **Individual** accounts.

Provisioning

This section is used to enable or disable basic device provisioning for this user if the **Starter Pack** option key is installed. Currently, provisioning is limited to registered Movi users.

By default, **Provision this user** is set to **On** which enables provisioning for the shown device URI.

The format of the **Provisioned device URI** is <username>.<device type>@<FindMe ID domain> and cannot be changed.



If [device authentication](#) using a local database is enabled, authentication credentials for the provisioned devices must also be set up in the local authentication database. The credential name must be the same as account username and the credential password must be the same as the password configured on the provisioned device.

Maintaining user accounts (continued)

Managing user accounts

After a user account has been created, you can configure additional details of that account.

From the **User accounts** page, click on the account you want to maintain. You are taken to the **Edit user account** page.

From here you can change:

- user details (**Display name**, **phone number**)
- the account's password (if using local authentication)
- FindMe details (**FindMe ID** and **FindMe type**)
- whether to provision the user (if the **Starter Pack** option key is installed)
- the devices and locations associated with that account
- the account's [Principal devices](#)

Configuring devices and locations

To add, modify or delete the FindMe devices and locations associated with a user account:

- Click **Edit user** in the **Configure devices and locations** section. This opens a new window from where you can add, modify or delete the devices and locations associated with the account.

Close the window when you have finished making changes.

Note that this is the same interface that users use when they log in to their own account to configure their FindMe details. See [Configuring your FindMe user account](#) for more information about how to use this interface.

Configuring principal devices

To configure the account's [Principal devices](#):

- Click **Edit principal devices** in the **Configure devices and locations** section.

This opens the **Edit principal devices** page which lists all of the devices currently associated with the selected user. The **Principal device** column indicates each device's current status as a principal device or not.

- To set devices as a principal device, select the box next to the required devices and click **Set as principal device**.
- To set devices so they are no longer principal devices, select the required devices and click **Unset as principal device**.

Changing an account's password

To change a password on behalf of a user without knowing their existing password:

- Enter the new password to be used when logging into this account into the **New password** and **Confirm password** fields and click **Save**.

This feature is useful if a user forgets their password.



If remote authentication is enabled, passwords are managed through your remote directory server instead.

Administrator and user groups

About administrator groups

The **Administrator groups** page lists all the administrator groups that have been configured on the VCS, and lets you add, edit and delete groups. Note that administrator groups are only active when the **Administrator authentication source** is set to **Remote**.

Administrator groups determine which access rights members of the group have after they have been successfully authenticated to use the VCS. The group **Name** defined in the VCS must match the group name that has been set up in the remote directory service to manage administrator access to this VCS.

When an administrator logs in to the VCS their credentials are authenticated against the remote directory service and they are assigned the access rights associated with the group to which the administrator belongs. If the administrator account belongs to more than one group, the highest level permission is assigned.

Configuring administrator groups

To go to the **Administrator groups** page:

- **Maintenance > Login accounts > Administrator groups.**

Click on the group you want to configure (or click **New** to create a new group, or click **Delete** to remove a group). You can create up to 30 administrator groups.

To configure administrator groups using the CLI:

- [xConfiguration Login Administrator Groups](#)

The configurable options are:

Name

The name of the administrator group.

Access

Read Write: allows all configuration to be viewed and changed. This provides the same rights as the default **admin** account.

Read Only: allows status and configuration information to be viewed only and not changed.

Auditor: allows access to the **Event Log**, **Configuration Log** and the **Overview** page only.

None: web login access to the VCS is not allowed.

About user groups

The **User groups** page lists all the user groups that have been configured on the VCS, and lets you add, edit and delete groups. You can create up to 15 user groups.

Note that user groups are only active when the **User authentication source** is set to **Remote**.

User groups determine which access rights members of the group have after they have been successfully authenticated to use the VCS. The group **Name** defined in the VCS must match the group name that has been set up in the remote directory service to manage user accounts.

When a user logs in to the VCS their credentials are authenticated against the remote directory service and they are assigned the access rights associated with the group to which that user belongs. If the user account belongs to more than one group, the highest level permission is assigned.

Configuring user groups

To go to the **User groups** page:

- **Maintenance > Login accounts > User groups.**

Click on the group you want to configure (or click **New** to create a new group, or click **Delete** to remove a group).

To configure user groups using the CLI:

- [xConfiguration Login User Groups](#)

The configurable options are:

Name

The name of the user group.

Access

Read Write: allows the user to view and change their FindMe details, devices and locations.

None: access to their user account is not allowed.

Group name limitations

Administrator and user group names cannot contain any of the following characters:

- / \ [] : ; | = , + * ? > < @ "

Account authentication using LDAP

Configuring LDAP server settings

The **Login account LDAP configuration** page is used to configure an LDAP connection to a remote directory service for administrator and/or user account authentication.

To go to the **Login account LDAP configuration** page:

- **Maintenance > Login accounts > LDAP configuration.**

To configure account LDAP settings using the CLI:

- [xConfiguration Login Remote LDAP](#)

LDAP server configuration

This section specifies the connection details to the LDAP server.

Server address

The IP address or FQDN (or server address, if a DNS **Domain Name** has also been configured) of the LDAP server hosting the database.

FQDN address resolution

If the **Server address** is an FQDN, this controls how it is resolved by the DNS server:

Address record: performs a DNS A or AAAA record lookup.

SRV record: performs a DNS SRV lookup. The advantage of using SRV records is that multiple (primary and backup) servers can be specified.

Port

The IP port to use on the LDAP server, typically 389 for non-TLS, and 636 if TLS encryption is enabled.

Encryption

Determines whether the connection to the LDAP server is encrypted using Transport Layer Security (TLS).

TLS: uses TLS Encryption for the connection to the LDAP server.

Off: no encryption is used.

The default is **Off**.

Certificate revocation list (CRL) checking

Specifies whether certificate revocation lists (CRLs) are checked when forming a TLS connection with the LDAP server.

Note that CRL data is uploaded to the VCS via the trusted CA certificate PEM file.

None: no CRL checking is performed.

Peer: only the CRL associated with the CA that issued the LDAP server's certificate is checked.

All: all CRLs in the trusted certificate chain of the CA that issued the LDAP server's certificate are checked.

The default is **None**.

Authentication configuration

This section specifies the VCS's authentication credentials to use when binding to the LDAP server.

VCS bind DN

The distinguished name used by the VCS when binding to the LDAP server.

VCS bind password

The password used by the VCS when binding to the LDAP server. The maximum plaintext length is 60 characters, which is then encrypted.

SASL

The SASL (Simple Authentication and Security Layer) mechanism to use when binding to the LDAP server.

None: no mechanism is used.

DIGEST-MD5: the DIGEST-MD5 mechanism is used.

The default is **DIGEST-MD5**.

VCS bind username

The username used by the VCS when binding to the LDAP server with SASL.

Directory configuration

This section specifies the base distinguished names to use when searching for account and group names.

Base DN for accounts

The distinguished name to use as the base when searching for administrator and user accounts.

Base DN for groups

The distinguished name to use as the base when searching for administrator and user groups.

Connection status

The current status of the connection to the specified LDAP server is displayed at the bottom of the page.



To use LDAP for account authentication, you must also go to the [Login account authentication configuration](#) page and select a **Remote** administrator or FindMe authentication source.

TLS encryption and CRL checking

The link **Upload a CA Certificate file for TLS** takes you to the **Security certificates** page, where you can upload a file containing the trusted CA certificate for the LDAP server. This is required if the connection between the VCS and the LDAP server is encrypted.

The CA certificate file should also contain any required CRL data.

See the [Security certificates](#) section for more information.



See the [VCS Deployment Guide - Authenticating VCS accounts using LDAP \[30\]](#) for more details on configuring an LDAP server, including help on specifying distinguished names for searching the database.



You can also use LDAP for device authentication. For more details, see [Device authentication using LDAP](#).

Root account

Root account

The VCS provides a root account which can be used to log in to the VCS operating system. This account has a username of **root** (all lower case) and a default password of **TANDBERG** (all upper case). For security reasons you must change the password as soon as possible. A warning is displayed on the web interface and the CLI if the **root** account has the default password set.



The **root** account should not be used in normal operation, and in particular system configuration should not be conducted using this account. Use the **admin** account instead.

Changing the root account password

To change the password for the **root** account:

1. Log in to the VCS as **root**. By default you can only do this using a serial connection or SSH.
2. Type **passwd**.
You will be asked for the new password.
3. Enter the new password and when prompted, retype the password.
4. Type **exit** to log out of the root account.

Accessing the root account over SSH and Telnet

By default, the root account can be accessed over a serial connection or SSH only - access over Telnet is disabled by default. You may wish to enable access over Telnet, but for security reasons we do not recommend this.

To enable and disable access to the root account using SSH and Telnet:

1. Log in to the VCS as **root**.
2. Type one of the following commands:
 - **rootaccess -t on** to enable access using Telnet
 - **rootaccess -t off** to disable access using Telnet
 - **rootaccess -s on** to enable access using SSH
 - **rootaccess -s off** to disable access using SSH
3. Type **exit** to log out of the root account.



If you have disabled SSH access while logged in using SSH, your current session will remain active until you log out, but all future SSH access will be denied.



If your VCS is part of a cluster, do not disable root access using SSH. The clustering feature depends on SSH access.

Resetting passwords

Resetting a forgotten administrator or root password

If you forget the password for the default **admin** account or any other administrator account, log in to the VCS using the account of another administrator with read/write access and change the password.

However, if you do not have any other administrator accounts with read/write access, or have forgotten the passwords for them all, you can set a new password for the **admin** account using the following procedure. This can also be used if you have forgotten the password for the **root** account:

1. Connect a PC to the VCS using the serial cable as per the instructions in the [VCS Getting Started Guide \[28\]](#).
2. Restart the VCS.
3. Log in from the PC with the username **pwrec**. No password is required.
4. Select the account (**root** or **admin**) whose password you wish to change.
5. You will be prompted for a new password.



The **pwrec** account is only active for one minute following a restart. After that time you will have to restart the system again to change the password.

Overview

About administration access

While you can administer the VCS by using a PC connected directly to the unit via a serial cable, you may want to access the system remotely over IP.

You can do this using either or both:

- the [web interface](#) via HTTPS
- a [command line interface](#) (CLI) via SSH or Telnet.

By default, access via HTTPS and SSH is enabled; access via Telnet is disabled. These can be enabled and disabled according to your requirements.

You can also enable access via HTTP. However, this mode works by redirecting HTTP calls to the HTTPS port, so HTTPS must be enabled for access via HTTP.



Cisco TelePresence Management Suite (Cisco TMS) accesses the VCS via the web server. If HTTPS mode is turned off, Cisco TMS will not be able to access it.

Security considerations

To securely manage the VCS you should disable Telnet, using the encrypted HTTPS and SSH protocols instead. For further security, disable HTTPS and SSH as well and use the serial port to manage the system.



Because access to the serial port allows the password to be reset, it is recommended that you install the VCS in a physically secure environment.

Configuring administration access

To configure the services by which your system can be accessed:

- **System configuration > System.**
You will be taken to the **System administration** page. In the **Admin Access** section, select **Off** or **On** from the drop-down boxes for each service.
- [xConfiguration Administration](#)



You must restart the system for any changes to the administration settings to take effect.

Administration session timeout

By default, administration sessions do not time out – they remain active until you log out.

You can set the system to time out an administration session after a set number of minutes of inactivity. The timeout period will apply to all administration sessions using both the web interface and the command line interface.

To set the timeout period:

- **System configuration > System.**
You will be taken to the **System administration** page. In the **Admin Access** section, in the **Session time out (minutes)** box, enter the number of minutes of inactivity after which an administration session should time out.
- [xConfiguration Administration TimeOut](#)

Values must be between 0 and 10,000. A value of 0 means that administration sessions will never time out.



You must restart the system for any changes to the administration settings to take effect.

Overview

The backup and restore features are used to create and restore backup files of your VCS data.

You are recommended to backup your data:

- before performing an upgrade or a system restore
- in demonstration and test environments if you want to be able to restore the VCS to a known configuration

You can independently backup and restore two sets of data:

System data, which includes:

- system configuration settings
- Call Policy
- clustering configuration
- security certificates
- administrator account details
- user accounts and FindMe settings (when the **Starter Pack** option key is installed)

TMS Agent data, which includes:

- user accounts and FindMe settings (when the **Starter Pack** option key is not installed)
- TMS Agent provisioning accounts and settings

Note that event logs are not included in the backup files.

Limitations

- Backups can only be restored to a VCS running the same version of software from which the backup was made.
- You can create a backup on one VCS and restore it to a different VCS, for example if the original system has failed. However, before performing the restore you must install on the new system the same set of option keys that were installed on the old system. If you do attempt to restore a backup made on a different VCS, you will receive a warning message, but you will be allowed to continue.
- Backups should not be used to copy data between VCSs.



For extra information about backing up and restoring peers in a cluster, refer to the [Backup and restore](#) section of the [Clustering and peers](#) chapter.

Creating a backup of your VCS data

System data

To create a backup of the VCS's **System** data:

1. Go to the **Backup and restore** page.
Maintenance > Backup and restore.
2. Click **Create system backup file.**
3. After the backup file has been prepared, a pop-up window appears and prompts you to save the file (the exact wording depends on your browser). The default name is in the format: **<hardware serial number> _ <date> _ <time> _ backup.tar.gz.**
4. Save the file to a designated location.

TMS Agent data

To create a backup of the VCS's TMS Agent data:

1. Go to the **Backup and restore** page.
Maintenance > Backup and restore.
2. Click **Create TMS Agent backup file.**
3. After the backup file has been prepared, a pop-up window appears and prompts you to save the file (the exact wording depends on your browser). The default name is in the format: **<date> _ <time> _ tms _ agent _ backup.tar.gz.**
Note that the preparation of the TMS Agent backup file may take several minutes. Do not navigate away from this page while the file is being prepared.
4. Save the file to a designated location.

Restoring a previous backup

System data



You are recommended to take the VCS unit out of service before performing a restore.

To restore the VCS to a previous configuration of **System** data:

1. Go to the **Backup and restore** page.
Maintenance > Backup and restore.
2. In the **Restore** section, **Browse** to the backup file containing the configuration you want to restore.
3. Click **Upload system backup file.**
4. The VCS checks the file and takes you to the **Restore confirmation** page.
 - If the backup file is not valid, you will receive an error message at the top of the **Backup and restore** page.You are shown the current software version and the number of calls and registrations.
5. Read all the **Warning** messages that appear before proceeding with the restore.
6. Click **Continue with system restore** to continue with the restore process. This will restart your system, so ensure that there are no active calls.
 - Click **Abort system restore** if you need to exit the restore process and return to the **Backup and restore** page.
7. After the system restart, you are taken to the login page.

TMS Agent data

To restore the VCS to a previous set of **TMS Agent** data:

1. Go to the **Backup and restore** page.
Maintenance > Backup and restore.
2. In the **Restore** section, **Browse** to the backup file containing the configuration you want to restore.
3. Click **Upload TMS Agent backup file.**
4. The VCS checks the file and restores its contents.
 - If the backup file is not valid, you will receive an error message at the top of the **Backup and restore** page.

System snapshot

Overview

The system snapshot is used for diagnostic purposes. It is a file that can be sent to your Cisco support representative at their request to assist them in troubleshooting issues you may be experiencing.

Creating a system snapshot

To create a system snapshot file:

- **Maintenance > System snapshot.**
You will be taken to the **System snapshot** page.

[Create system snapshot](#)

Clicking this button starts the download of the system snapshot file. You will then be asked whether and where you would like to save the file.

Select a location from which you can easily send the file to your Cisco support representative.

Overview

The incident reporting feature of the VCS automatically saves information about critical system issues such as application failures. You can:

- [view the reports](#) from the VCS web interface
- [download and send the reports manually](#) to Cisco (usually at the request of Cisco customer support)
- configure the VCS to [send the reports automatically](#) to Cisco

The information contained in these reports can then be used by Cisco customer support to diagnose the cause of the failures. All information gathered during this process will be held in confidence and used by Cisco personnel for the sole purpose of issue diagnosis and problem resolution.

This feature is only intended for use at the request of Cisco customer support in exceptional situations, and is off by default.

Warning: privacy-protected personal data



IN NO EVENT SHOULD PRIVACY-PROTECTED PERSONAL DATA BE INCLUDED IN ANY REPORTS TO CISCO. Privacy-Protected Personal Data means any information about persons or entities that the Customer receives or derives in any manner from any source that contains any personal information about prospective, former, and existing customers, employees or any other person or entity. Privacy-Protected Personal Data includes, without limitation, names, addresses, telephone numbers, electronic addresses, social security numbers, credit card numbers, customer proprietary network information (as defined under 47 U.S.C. § 222 and its implementing regulations), IP addresses or other handset identifiers, account information, credit information, demographic information, and any other information that, either alone or in combination with other data, could provide information specific to a particular person.

PLEASE BE SURE THAT PRIVACY-PROTECTED PERSONAL DATA IS NOT SENT TO CISCO WHEN THE VCS IS CONFIGURED TO AUTOMATICALLY SEND REPORTS.

IF DISCLOSURE OF SUCH INFORMATION CANNOT BE PREVENTED, PLEASE DO NOT USE THE AUTOMATIC CONFIGURATION FEATURE. Instead, copy the data from the **Incident detail** page and paste it into a text file. You can then edit out any sensitive information before forwarding the file on to Cisco customer support. See the section [Sending incident reports manually](#) for more information.

What information does the report contain?

The information contained in the report is:

Time	the date and time at which the incident occurred.
Version	the version of software running when the incident occurred.
Build	the internal build number for the software.
Name	the name of the software.
System	the configured system name.
Serial number	the hardware serial number.
Process ID	the process ID the VCS application had when the incident occurred.
Release	a True/False flag indicating if this is release build (rather than a development build).
User name	the name of the person that built this software. This is blank for release builds.
Stack	the trace of the thread of execution that caused the incident.
Debug information	a full trace of the application call stack for all threads and the values of the registers.



For each call stack the **Debug information** will include the contents of variables which may contain some sensitive information, for example alias values and IP addresses. If your deployment is such that this information could contain information specific to a particular person, please read the **Warning: privacy-protected personal data** section (left) before you decide whether to enable automatic incident reporting.

If you choose not to enable automatic incident reporting, you can still manually send the reports to Cisco customer support. See the [Sending incident reports manually](#) section for more information.

Viewing incident reports

Viewing a list of all incidents

To view a list of all incidents that have occurred since the system was last upgraded:

- **Maintenance > Incident reporting > View.**
You will be taken to the **Incident view** page.
- [xConfiguration Error Reports](#)

Time

The date and time at which the incident occurred.

Version

The VCS software version running at the time the incident occurred.

Build

The build number of the VCS software version running at the time the incident occurred.

State

The current state of the incident.

Pending: the incident has been saved locally but not sent.

Sent: details of the incident have been sent to the URL specified in the **Incident reporting configuration** page.

Related tasks

To enable and disable automatic incident reporting, and configure the URL to which the reports are sent, click the **Reconfigure incident reporting** link. This takes you to the **Incident reporting configuration** page.

Viewing details of a particular incident

To view the information contained in the report for a particular incident report, click on the report's **Time**. You are taken to the **Incident detail** page. From here you can view the report on screen, or download it as an XML file for forwarding manually to Cisco customer support.

Sending incident reports manually



Please read the section [Warning: privacy-protected personal data](#) on the previous page before you decide whether to send an incident report manually to Cisco.

To send an incident report manually to Cisco customer support:

1. **Maintenance > Incident reporting > View.**
You will be taken to the **Incident view** page.
2. Click on the incident you wish to send.
You will be taken to the **Incident detail** page.
3. Scroll down to the bottom of the page and click **Download incident report**.
You will be given the option to save the file.
4. Save the file in a location from which it can be forwarded to Cisco customer support.

Removing sensitive information from the report

The details in the downloaded XML file are Base64-encoded, so you will not be able to meaningfully view or edit the information in the file accessed using the **Download incident report** button.

If you wish to edit the report before sending it to Cisco (for example, if you wish to remove any potentially sensitive information) you must copy and paste the information from the **Incident detail** page into a text file, and edit the information in the file before sending it to Cisco.

Sending incident reports automatically



Please read the section [Warning: privacy-protected personal data](#) on the previous page before you decide whether to enable automatic incident reporting.

To configure the VCS to send incident reports automatically to Cisco customer support:

- **Maintenance > Incident Reporting > Configuration.**
You will be taken to the **Incident Reporting Configuration** page.
- [xConfiguration Error Reports](#)

The options are:

Incident reports sending mode

Determines whether the VCS will automatically send details of application failures to a specified web service.

On: reports will be automatically sent to the URL specified below.

Off: incidents will not be sent to any URL but they will still be saved locally and can be viewed from the **Incident View** page.

Incident reports URL

If **Incident reports sending mode** is **On**, this is the URL of the web service to which any error reports will be sent.

Related tasks

To view a list of all existing incidents, click on the **View incidents** link. This will take you to the **Incident View** page.

Check pattern

The **Check pattern** page (**Maintenance > Tools > Check pattern**) allows you to test whether a pattern or transform you intend to configure on the VCS will have the expected result.

Patterns can be used when configuring:

- [Allow Lists and Deny Lists](#), to specify aliases to be included in the list
- [Pre-search transforms](#), to specify aliases to be transformed before any searches take place
- [Search rules](#) to filter searches based on the alias being searched for, and to transform an alias before the search is sent to that zone
- [Subzone membership rules](#) to determine, based on the address of the device, to which subzone an endpoint is assigned when it registers with the VCS

To use this tool:

Alias

Enter the alias against which you will test the transform.

Pattern

In the **Pattern** section, enter the combination of **Pattern type** and **Pattern behavior** for the **Pattern string** being tested.

- If you select a **Pattern behavior** of **Replace**, you also need to enter a **Replace** string.
- If you select a **Pattern behavior** of **Add prefix** or **Add suffix**, you also need to enter an **Additional text** string to append/prepend to the **Pattern string**.

Click **Check pattern** to test whether the alias matches the pattern.

The **Result** section shows whether the alias matched the pattern, and displays the transformed alias if appropriate.

Locate

The **Locate** page (**Maintenance > Tools > Locate**) lets you test whether the VCS can find an endpoint identified by a given alias, within a specified number of "hops", without actually placing a call to that endpoint. The results include the list of zones that were searched, any transforms and Call Policy that were applied, and if found, the zone in which the alias was located.

This tool is useful when diagnosing dial plan and network deployment issues.

To use this tool:

Alias

Enter the alias you want to locate.

Hop count

Enter the number of hops for the search.

Protocol

The protocol used to initiate the search, either H.323 or SIP. The search may be interworked during the search process, but the VCS always uses the native protocol first to search those target zones associated with search rules at the same priority, before searching those zones again using the alternative protocol.

Source zone

The zone from which to simulate the search request. Choose from the **Default Zone** (an unknown remote system), the **Local Zone** (a locally registered endpoint) or any other configured neighbor, traversal client or traversal server zone.

Click **Locate** to start the search. The status bar shows **Searching...** followed by **Search completed**. The body of the status section then shows whether or not the alias was found, any transforms that were applied, and the zones that were searched in order of priority.



The **Locate** tool performs the search as though the VCS received a call request from the selected **Source zone**. For more information, see the [Zone searching and transform process](#) section.

Port usage

Overview

The pages under the **Maintenance > Tools > Port usage** menu show, in table format, all the IP ports that have been configured on the VCS. The information shown on these pages is specific to that particular VCS and will vary depending on the VCS's configuration, the option keys that have been installed and the features that have been enabled.

The information can be sorted according to any of the columns on the page, so for example you can sort the list by IP port, or by IP address.

Each page contains an **Export to CSV** option. This lets you save the information in a CSV (comma separated values) format file suitable for opening in a spreadsheet application.



IP ports cannot be configured separately for IPv4 and IPv6 addresses, nor for each of the two LAN interfaces - in other words, after an IP port has been configured for a particular service, e.g. SIP UDP, this will apply to all IP addresses of that service on the VCS. Because the tables on these pages list all IP ports and all IP addresses, a single IP port may appear on the list up to 4 times, depending on your VCS configuration.

The port information is split into three pages, each described in the sections that follow.

Further information can also be found in the [Port reference](#) section.

Local VCS inbound ports

This page shows the listening ports on this VCS. These are the IP ports on the VCS used to receive inbound communications from other systems.

For each port listed on this page, if there is a firewall between the VCS and the source of the inbound communications, your firewall must allow:

- inbound traffic to the IP port on the VCS from the source of the inbound communications, and
- return traffic from that same VCS IP port back out to the source of the inbound communication.



This firewall configuration is particularly important if this VCS is a traversal client or traversal server, in order for Expressway firewall traversal to function correctly. See the [About Expressway™](#) section for more information.

Local VCS outbound ports

This page shows the source IP ports used by this VCS. These are the IP ports on the VCS used to send outbound communications to other systems.

For each port listed on this page, if there is a firewall between the VCS and the destination of the outbound communications, your firewall must allow:

- outbound traffic out from the IP port on the VCS to the destination of the outbound communications, and
- return traffic from that destination back to the same VCS IP port.



This firewall configuration is particularly important if this VCS is a traversal client or traversal server, in order for Expressway firewall traversal to function correctly. See the [About Expressway™](#) section for more information.

Remote listening ports

This page shows the destination IP addresses and IP ports of remote systems with which the VCS communicates.

Your firewall must be configured to allow traffic originating from the local VCS to the remote devices identified by the IP addresses and IP ports listed on this page.



There are other remote devices not listed here to which the VCS will be sending media and signaling, but the ports on which these devices receive traffic from the VCS is determined by the configuration of the destination device, so they cannot be listed here. If you have opened all the ports listed in the **Local outbound ports** page, the VCS will be able to communicate with all remote devices. You only need to use the information on this page if you wish to limit the IP ports opened on your firewall to these remote systems and ports.

Restarting



Do not restart the VCS while the red ALM LED on the front of the box is on. This indicates a hardware fault. Contact your Cisco representative.

The restart function shuts down and restarts the VCS application software, but not the operating system or hardware.

Some configuration changes require a restart of the VCS before they take effect. A **Restart** button is at the bottom of any web page that includes such settings, and clicking it takes you to the **Restart** page. A system warning will remain in place until the system is restarted.

Restarting causes any active calls and registrations to be terminated. For this reason, the **Restart** page displays the number of current calls and registrations, so you can check these before you restart.



If you do not restart the system immediately, you should refresh this page before restarting to check the current status of calls and registrations.

Restarting using the web interface

To restart the VCS using the web interface:

1. Go to **Maintenance > Restart**, or from a relevant configuration page, click the **Restart** button. You are taken to the **Restart** page.
2. Check the number of calls and registrations currently in place.
3. Click **Restart system**.

The **Restarting** page appears, with an orange bar indicating progress.

After the system has successfully restarted, you are automatically taken to the **Login** page.

Restarting using the CLI

To restart the VCS using the CLI:

- [xCommand Restart](#)

Rebooting



Do not reboot the VCS while the red ALM LED on the front of the box is on. This indicates a hardware fault. Contact your Cisco representative.

The reboot function shuts down and restarts the VCS application software, operating system and hardware.

Reboots are normally only required after software upgrades and are performed as part of the upgrade process.

Rebooting causes any active calls and registrations to be terminated. For this reason, the **Reboot** page displays the number of current calls and registrations, so you can check these before you reboot.



If you do not reboot the system immediately, you should refresh this page before rebooting to check the current status of calls and registrations.

Rebooting using the web interface

To reboot the VCS using the web interface:

1. Go to **Maintenance > Reboot**. You are taken to the **Reboot** page.
2. Check the number of calls and registrations currently in place.
3. Click **Reboot system**.

The **Rebooting** page appears, with an orange bar indicating progress.

After the system has successfully rebooted, you are automatically taken to the **Login** page.

Rebooting using the CLI

To reboot the VCS using the CLI:

- [xCommand Boot](#)

Shutting down



Do not shut down the system while the red ALM LED on the front of the box is on. This indicates a hardware fault. Contact your Cisco representative.



The system must be shut down before it is unplugged.



After the system has been shut down, the only way it can be restarted is by pressing the soft power button on the unit itself. You must therefore have physical access to the unit if you want to restart it after it has been shut down.

Shutting down causes any active calls and registrations to be terminated. For this reason, the **Shutdown** page displays the number of current calls and registrations, so you can check these before you shut down.



If you do not shut down the system immediately, you should refresh this page before shutting down to check the current status of calls and registrations.

Shutting down using the web interface

To shut down the VCS:

1. Go to **Maintenance > Shutdown**. You are taken to the **Shutdown** page.
2. Check the number of calls and registrations currently in place.
3. Click **Shutdown System**.

The **Shutting down** page appears. This page remains in place after the system has successfully shut down but any attempts to refresh the page or access the VCS will be unsuccessful.

Shutting down using the CLI

The VCS cannot be shut down using the CLI.

Overview

It is possible to restore the VCS to its default configuration. This is done through the CLI using **xCommand DefaultValuesSet**. This command is not available through the web interface.

The **DefaultValuesSet** command allows you to specify the level of configuration to restore, from 1 to 3 as follows:

- **Level 1** resets most configuration items to their default value, with the exception of the Level 2 and Level 3 items shown in the tables below.
- **Level 2** resets configuration items mostly related to remote authentication (listed in [Configuration items reset by DefaultValuesSet level 2](#)), plus Level 1 items to their default values.
- **Level 3** resets all critical configuration items (listed in [Configuration items reset by DefaultValuesSet level 3](#) below) plus Level 1 and Level 2 items to their default values.

The [xConfiguration reference](#) table shows a full list of all configuration items and where applicable their default values.



xCommand DefaultValuesSet Level: 3 must be used with caution, as it resets the system's IPv4 and IPv6 addresses, meaning you will no longer be able to access the system over IP. It also deletes all option keys including pre-installed options such as Expressway and the number of calls. It also deletes all links configured on the VCS, including the automatically configured [default links](#) between the Default Subzone, Traversal Subzone and Default Zone. Without these links, calls will not be able to be placed. To restore these links, you should run the command **xCommand DefaultLinksAdd** after **xCommand DefaultValuesSet Level: 3**. These links can also be restored manually using the web interface - see the [Creating and editing links](#) section for instructions on how to do this.

Configuration items reset by DefaultValuesSet level 3

Configuration item	Default value after xCommand DefaultValuesSet Level: 3
Administration HTTP Mode	On
Administration HTTPS Mode	On
Administration SSH Mode	On
Administration Telnet Mode	Off
Ethernet [1..2] IP V4 Address	192.168.0.100
Ethernet [1..2] IP V4 StaticNAT Address	<blank>
Ethernet [1..2] IP V4 StaticNAT Mode	Off
Ethernet [1..2] IP V4 SubnetMask	255.255.255.0
Ethernet [1..2] IP V6 Address	<blank>
Ethernet [1..2] Speed	Auto
IPProtocol	IPv4
IP DNS Domain Name	<blank>
IP DNS Hostname	<blank>
IP DNS Server [1..5] Address	<blank>
IP Gateway	127.0.0.1
IP Route [1..50] Address	<blank>

Restoring default configuration

Configuration items reset by DefaultValuesSet level 3 (cont.)

Configuration item	Default value after xCommand DefaultValuesSet Level: 3
IP Route [1..50] Gateway	<blank>
IP Route [1..50] Interface	Auto
IP Route [1..50] PrefixLength	32
IP V6 Gateway	<blank>
NTP Address	<blank>
Option [1..64] Key	<all option keys are deleted>
SystemUnit AdminAccount [1..15] Access	ReadWrite
SystemUnit AdminAccount [1..15] Name	<blank>
SystemUnit AdminAccount [1..15] Password	<blank>
SystemUnit Maintenance Mode	Off
SystemUnit Name	<blank>
SystemUnit Password	TANDBERG
SystemUnit StrictPassword Enforce	Off

Configuration items reset by DefaultValuesSet level 2

Configuration item	Default value after xCommand DefaultValuesSet Level: 2
Alternates Cluster Name	<blank>
Login Administrator Groups Group [1..30] Access	ReadWrite
Login Administrator Groups Group [1..30] Name	<blank>
Login Administrator Source	Local
Login Remote LDAP BaseDN Accounts	<blank>
Login Remote LDAP BaseDN Groups	<blank>
Login Remote LDAP DirectoryType	ActiveDirectory
Login Remote LDAP Encryption	Off
Login Remote LDAP SASL	DIGEST-MD5

Restoring default configuration

Configuration items reset by DefaultValueSet level 2 (cont.)

Configuration item	Default value after xCommand DefaultValueSet Level: 2
Login Remote LDAP Server Address	<blank>
Login Remote LDAP Server Port	389
Login Remote LDAP VCS BindDN	<blank>
Login Remote LDAP VCS BindPassword	<blank>
Login Remote LDAP VCS BindUsername	<blank>
Login Remote Protocol	LDAP
Login User Groups Group [1..15] Access	ReadWrite
Login User Groups Group [1..15] Name	<blank>
Login User Source	Local

Overview

All passwords configured on the VCS are stored in encrypted form. This applies to the following, which all have usernames and passwords associated with them:

- the default admin administrator account
- any additional administrator accounts
- local authentication database credentials (a list of valid usernames and passwords that are used when other devices are required to authenticate with the VCS)
- outbound connection credentials (used by the VCS when required to authenticate with another system)
- LDAP server (used by the VCS when binding to an LDAP server)

Passwords can be configured using either the CLI or the web interface.

Web interface

When entering or viewing passwords using the web interface, you will see placeholder characters (e.g. dots or stars, depending on your browser) instead of the characters you are typing.

Command line interface (CLI)

When entering passwords using the command line interface (CLI), you will type the password in plain text. However, after the command has been executed, the password will be displayed in its encrypted form with a **{cipher}** prefix, e.g.

xConfiguration LDAP Password: "{cipher}xcy6k+4NgB025vYEgoEXXw=="



FindMe is a standalone application that can be hosted by the VCS or by another remote server. This means that FindMe user account information is not configured or accessible using the CLI of the VCS. However, FindMe user passwords are still stored securely.

Maximum length of passwords

When a password is encrypted, it uses more characters than the original plain text version of the password. For each type of password, the maximum number of plain text characters that can be entered and the maximum number of encrypted characters that are displayed through the CLI are shown in the table below.

Password type	Maximum plain text characters	Maximum displayed encrypted characters
Admin account	16	65
Administrator accounts	16	65
Local Database authentication credentials	128	215
Outbound connection credentials	128	215
LDAP server	60	122
FindMe accounts	30	82

Appendices

This section includes the following appendices which provide supplementary information regarding the administration of the Cisco VCS:

- [CPL reference](#)
- [Regular expression reference](#)
- [Pattern variable reference](#)
- [Port reference](#)
- [DNS configuration](#)
- [LDAP configuration for device authentication](#)
- [xConfiguration command reference](#)
- [xCommand command reference](#)
- [xStatus command reference](#)
- [VCS warnings](#)
- [Bibliography](#)
- [Glossary](#)
- [Legal notices](#)



Overview of CPL on the Cisco VCS

Call Processing Language (CPL) is an XML-based language for defining call handling. This Appendix gives details of the VCS's implementation of the CPL language and should be read in conjunction with the CPL standard [RFC 3880 \[5\]](#) and the [Guide to writing CPL \[22\]](#).

The VCS has many powerful inbuilt transform features so CPL should be required only if advanced call handling rules are required.

The VCS supports most of the CPL standard along with some TANDBERG-defined extensions. It does not support the top level actions `<incoming>` and `<outgoing>` as described in RFC 3880. Instead it supports a single section of CPL within a `<taa:routed>` section.

When Call Policy is implemented by uploading a CPL script to the VCS, the script is checked against an XML schema to verify the syntax. There are two schemas - one for the basic CPL specification and one for the TANDBERG extensions. Both these schemas can be [downloaded from the web interface](#) and used to validate your script before uploading to the VCS.

The following example shows the correct use of namespaces to make the syntax acceptable:

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
  xmlns:taa="http://www.tandberg.net/cpl-extensions"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
<taa:routed>
  <address-switch field="destination">
    <address is="reception@example.com">
      <proxy/>
    </address>
  </address-switch>
</taa:routed>
</cpl>
```

address-switch

Overview

The **address-switch** node allows the script to run different actions based on the source or destination aliases of the call. It specifies which fields to match, and then a list of **address** nodes contains the possible matches and their associated actions.

The **address-switch** has two node parameters: [field](#) and [subfield](#).

address

The **address** construct is used within an **address-switch** to specify addresses to match. It supports the use of Regular Expressions (see the [Regular expression reference](#) Appendix for further information).

Valid values are:

is=string	Selected field and subfield exactly match the given string.
contains=string	Selected field and subfield contain the given string. Note: The CPL standard only allows for this matching on the display subfield; however the VCS allows it on any type of field.
subdomain-of=string	If the selected field is numeric (e.g. the tel subfield) then this matches as a prefix; so address subdomain-of="555" matches 5556734 etc. If the field is not numeric then normal domain name matching is applied; so address subdomain-of="company.com" matches nodeA.company.com etc.
regex="regular expression"	Selected field and subfield match the given regular expression.

All address comparisons ignore upper/lower case differences so **address is="Fred"** will also match **fred**, **freD** etc.

address-switch

field

Within the [address-switch node](#), the mandatory **field** parameter specifies which address is to be considered. The supported attributes and their interpretation are as follows:

	Authentication mode: On		Authentication mode: Off	
Field parameter attributes	SIP	H.323	SIP	H.323
origin	The "From" and "ReplyTo" fields of the message if it authenticated correctly, otherwise not-present .	The source aliases from the original LRQ or ARQ that started the call if it authenticated correctly, otherwise not-present . Because SETUP messages are not authenticated, if the VCS receives a SETUP without a preceding RAS message the origin will always be not-present .	The "From" and "ReplyTo" fields of the incoming message.	The source aliases from the original LRQ or ARQ that started the call. If a SETUP is received without a preceding RAS message then the origin is taken from the SETUP.
unauthenticated-origin	The "From" and "ReplyTo" fields of the incoming message.	The source aliases from the original LRQ or ARQ that started the call. If a SETUP is received without a preceding RAS message then the origin is taken from the SETUP.	The "From" and "ReplyTo" fields of the incoming message.	The source aliases from the original LRQ or ARQ that started the call. If a SETUP is received without a preceding RAS message then the origin is taken from the SETUP.
authenticated-origin	The "From" and "ReplyTo" fields of the message if it authenticated correctly, otherwise not-present .	The source aliases from the original LRQ or ARQ that started the call if it authenticated correctly otherwise empty. Because SETUP messages are not authenticated, if the VCS receives a SETUP without a preceding RAS message the origin will always be not-present .	not-present	
originating-zone	The name of the zone or subzone for the originating leg of the call. If the call originates from a neighbor, traversal server or traversal client zone then this will equate to the zone name. If it comes from an endpoint within one of the local subzones this will be the name of the subzone. If the call originates from any other locally registered endpoint this will be " DefaultSubZone ". In all other cases this will be " DefaultZone ".			
originating-user	The username used for authentication.		not-present	
registered-origin	If the call originates from a registered endpoint this is the list of all aliases it has registered, otherwise not-present .			
destination	The destination aliases.			
original-destination	The destination aliases.			

If the selected field contains multiple aliases then the VCS will attempt to match each address node with all of the aliases before proceeding to the next address node i.e. an address node matches if it matches any alias.

address-switch

subfield

Within the [address-switch node](#), the optional **subfield** parameter specifies which part of the address is to be considered. The following table gives the definition of subfields for each alias type. If a subfield is not specified for the alias type being matched then the **not-present** action will be taken.

address-type	Either h323 or sip , based on the type of endpoint that originated the call.
user	For URI aliases this selects the username part. For H.323 IDs it is the entire ID and for E.164 numbers it is the entire number.
host	For URI aliases this selects the domain name part. If the alias is an IP address then this subfield is the complete address in dotted decimal form.
tel	For E.164 numbers this selects the entire string of digits.
alias-type	Gives a string representation of the type of alias. The type is inferred from the format of the alias. Possible types are: • Address Type • Result • URI • url-ID • H.323 ID • h323-ID • Dialed Digits • dialedDigits

otherwise

The **otherwise** node will be executed if the address specified in the **address-switch** was found but none of the preceding address nodes matched.

not-present

The **not-present** node is executed when the address specified in the **address-switch** was not present in the call setup message. This form is most useful when authentication is being used. With authentication enabled the VCS will only use authenticated aliases when running policy so the **not-present** action can be used to take appropriate action when a call is received from an unauthenticated user (see the example [Call screening of authenticated users](#)).

location

As the CPL script is evaluated it maintains a list of addresses (H.323 IDs, URLs and E.164 numbers) which will be used as the destination of the call if a **proxy** node is executed. The **taa:location** node allows the location set to be modified so that calls can be redirected to different destinations.

At the start of script execution the location set is initialized to the original destination.

The following attributes are supported on **taa:location** nodes. It supports the use of Regular Expressions (see the [Regular expression reference](#) Appendix for further information).

Clear = "yes" "no"	Specifies whether to clear the current location set before adding the new location. The default is to append this location to the end of the set.
url=string	The new location to be added to the location set. The given string can specify a URL (e.g. user@domain.com), H.323 ID or an E.164 number.
priority =<0.0..1.0> "random"	Specified either as a floating point number in the range 0.0 to 1.0, or random , which assigns a random number within the same range. 1.0 is the highest priority. Locations with the same priority are searched in parallel.
regex =<regular expression> replace =<string>	Specifies the way in which a location matching the regular expression is to be changed.

rule-switch

This extension to CPL is provided to simplify Call Policy scripts that need to make decisions based on both the source and destination of the call. A **taa:rule-switch** can contain any number of rules that are tested in sequence; as soon as a match is found the CPL within that rule element is executed. Each rule must take one of the following forms:

```
<taa:rule-switch>
  <taa:rule origin="<regular expression>" destination="<regular expression>" message-regex="<regular expression>">
  <taa:rule authenticated-origin="<regular expression>" destination="<regular expression>" message-regex="<regular expression>">
  <taa:rule unauthenticated-origin="<regular expression>" destination="<regular expression>" message-regex="<regular expression>">
  <taa:rule registered-origin="<regular expression>" destination="<regular expression>" message-regex="<regular expression>">
  <taa:rule originating-user="<regular expression>" destination="<regular expression>" message-regex="<regular expression>">
  <taa:rule originating-zone="<regular expression>" destination="<regular expression>" message-regex="<regular expression>">
</taa:rule-switch>
```

The meaning of the various **origin** selectors is as described in the [field](#) section.

The **message-regex** parameter allows a regular expression to be matched against the entire incoming SIP message.



Note that any rule containing a **message-regex** parameter will never match an H.323 call.

proxy

On executing a **proxy** node the VCS attempts to forward the call to the locations specified in the current location set. If multiple entries are in the location set then this results in a forked call. If the current location set is empty the call is forwarded to its original destination.

The proxy node supports the following optional parameters:

timeout=<1..86400>	Timeout duration, specified in seconds
stop-on-busy = "yes" "no"	Whether to stop searching if a busy response is received

The proxy action can lead to the results shown in the table below.

failure	The proxy failed to route the call
busy	Destination is found but busy
noanswer	Destination is found but does not answer
redirection	VCS is asked to redirect the call
default	CPL to run if the other results do not apply

The CPL can perform further actions based on these results. Any results nodes must be contained within the **proxy** node. For example:

```
<proxy timeout="10">
  <busy>
    <!--If busy route to recording service-->
    <location clear="yes" url="recorder">
      <proxy/>
    </location>
  </busy>
</proxy>
```

reject

If a **reject** node is executed the VCS stops any further script processing and rejects the current call.

The custom reject strings **status=string** and **reason=string** options are supported here and should be used together to ensure consistency of the strings.

Unsupported CPL elements

The VCS does not currently support some elements that are described in the CPL RFC. If an attempt is made to upload a script containing any of the following elements an error message will be generated and the VCS will continue to use its existing policy.

The following elements are not currently supported:

- time-switch
- string-switch
- language-switch
- priority-switch
- redirect
- mail
- log
- subaction
- lookup
- remove-location

CPL examples

Call screening of authenticated users

In this example, only calls from users with authenticated source addresses are allowed. See the [Device authentication](#) section for details on how to enable authentication.

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
  xmlns:taa="http://www.tandberg.net/cpl-extensions"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <address-switch field="origin">
      <not-present>
        <!-- Reject call with a status code of 403 (Forbidden) -->
        <reject status="403" reason="Denied by policy"/>
      </not-present>
    </address-switch>
  </taa:routed>
</cpl>
```

Call screening based on alias

In this example, user ceo will only accept calls from users *vpsales*, *vpmarketing* or *vpengineering*.

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
  xmlns:taa="http://www.tandberg.net/cpl-extensions"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <address-switch field="destination">
      <address is="ceo">
        <address-switch field="origin">
          <address regex="vpsales|vpmarketing|vpengineering">
            <!-- Allow the call -->
            <proxy/>
          </address>
        </address-switch>
      </address>
    </address-switch>
  </taa:routed>
</cpl>
```

CPL examples

Call screening based on domain

In this example, user *fred* will not accept calls from anyone at *annoying.com*, or from any unauthenticated users. All other users will allow any calls.

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
  xmlns:taa="http://www.tandberg.net/cpl-extensions"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <address-switch field="destination">
      <address is="fred">
        <address-switch field="origin" subfield="host">
          <address subdomain-of="annoying.com">
            <!-- Don't accept calls from this source -->
            <!-- Reject call with a status code of 403 (Forbidden) -->
            <reject status="403" reason="Denied by policy"/>
          </address>
        </address-switch>
      </address>
    </address-switch>
  </taa:routed>
</cpl>
```

Change of domain name

In this example, Example Inc has changed its domain from *example.net* to *example.com*. For a period of time some users are still registered at *example.net*. The following script would attempt to connect calls to *user@example.com* first and if that fails then fallback to *example.net*.

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
  xmlns:taa="http://www.tandberg.net/cpl-extensions"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <address-switch field="destination">
      <address regex="(.*)@example.com">
        <proxy>
          <failure>
            <!-- Failed to contact using example.com, retry the request with
            example.net -->
            <taa:location clear="yes" regex="(.*)@example.com"
            replace="\1@example.net">
              <proxy/>
            </taa:location>
          </failure>
        </proxy>
      </address>
    </address-switch>
  </taa:routed>
</cpl>
```

CPL examples

Allow calls from locally registered endpoints only

In this example, the administrator only wants to allow calls that originate from locally registered endpoints.

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
    xmlns:taa="http://www.tandberg.net/cpl-extensions"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <address-switch field="registered-origin">
      <not-present>
        <reject status="403" reason="Only local endpoints can use this VCS"/>
      </not-present>
    </address-switch>
  </taa:routed>
</cpl>
```

Block calls from Default Zone and Default Subzone

The same script can be extended to also allow calls from configured zones but not from the Default Zone or Default Subzone.

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
    xmlns:taa="http://www.tandberg.net/cpl-extensions"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <address-switch field="registered-origin">
      <not-present>
        <address-switch field="originating-zone">
          <address is="DefaultZone">
            <!-- Reject call with a status code of 403 (Forbidden) -->
            <reject status="403" reason="Denied by policy"/>
          </address>
          <address is="DefaultSubZone">
            <!-- Reject call with a status code of 403 (Forbidden) -->
            <reject status="403" reason="Denied by policy"/>
          </address>
          <otherwise>
            <proxy/>
          </otherwise>
        </address-switch>
      </not-present>
    </address-switch>
  </taa:routed>
</cpl>
```

CPL examples

Restricting access to a local gateway

In these examples, a gateway is registered to the VCS with a prefix of 9 and the administrator wants to stop calls from outside the organization being routed through it.

This can be done in two ways: using the `address-switch` node or the `taa:rule-switch` node. Examples of each are shown below.

Using the address-switch node

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
  xmlns:taa="http://www.tandberg.net/cpl-extensions"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <address-switch field="destination">
      <address regex="9(.*)">
        <address-switch field="originating-zone">
          <!-- Calls coming from the traversal zone are not allowed to use this
gateway -->
          <address is="TraversalZone">
            <!-- Reject call with a status code of 403 (Forbidden) -->
            <reject status="403" reason="Denied by policy"/>
          </address>
        </address-switch>
      </address>
    </address-switch>
  </address>
</address-switch>
</taa:routed>
</cpl>
```

Using the taa:rule-switch node

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
  xmlns:taa="http://www.tandberg.net/cpl-extensions"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <taa:rule-switch>
      <taa:rule originating-zone="TraversalZone" destination="9(.*)">
        <!-- Calls coming from the traversal zone are not allowed to use this
gateway -->
        <!-- Reject call with a status code of 403 (Forbidden) -->
        <reject status="403" reason="Denied by policy"/>
      </taa:rule>
      <taa:rule origin="(*)" destination="(*)">
        <!-- All other calls allowed -->
        <proxy/>
      </taa:rule>
    </taa:rule-switch>
  </taa:routed>
</cpl>
```

CPL examples

Redirecting failed calls based on status code

The output from a **proxy** node allow actions to be taken based on the result of the proxy operation. In base CPL a single failure output is allowed which will be invoked if the call attempt fails for any reason (see section 6.1 of [RFC 3880 \[5\]](#) for details).

The VCS supports an extension to the base CPL specification that allows a status code to be specified so that the failure action is only invoked if the call attempt fails for the specified reason. In addition the VCS allows multiple failure outputs to be specified within a single **proxy** node. This allows a script to redirect the call to different locations (e.g. different recorded messages) based on the exact reason for call failure.

For example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
  xmlns:taa="http://www.tandberg.net/cpl-extensions"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <!-- Proxy the call normally, but redirect to different recorded
messages based on -->
    <!-- the particular error response we get -->
    <proxy>
      <failure status="403">
        <!-- Call attempt failed with 403 (Forbidden) -->
        <taa:location url="forbidden-message@example.com" clear="yes">
          <proxy/>
        </taa:location>
      </failure>
      <failure status="404">
        <!-- Call attempt failed with 404 (Not Found) -->
        <taa:location url="notfound-message@example.com" clear="yes">
          <proxy/>
        </taa:location>
      </failure>
    </proxy>
  </taa:routed>
</cpl>
```

```
<failure>
  <!-- General catch-all failure handler for all other error responses
-->
  <taa:location url="failed-message@example.com" clear="yes">
    <proxy/>
  </taa:location>
</failure>
</proxy>
</taa:routed>
</cpl>
```

CPL examples

Reject attempts to subscribe to a presentity

In this example, attempts to subscribe the presence of *user@example.com* are rejected.

```
<?xml version="1.0" encoding="UTF-8" ?>
<cpl xmlns="urn:ietf:params:xml:ns:cpl"
      xmlns:taa="http://www.tandberg.net/cpl-extensions"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="urn:ietf:params:xml:ns:cpl cpl.xsd">
  <taa:routed>
    <taa:rule-switch>
      <taa:rule origin="*" destination="user@example.com"
        message-regex="^SUBSCRIBE.*">
        <!-- Cannot subscribe to user@example.com -->
        <!-- Reject call with a status code of 403 (Forbidden) -->
        <reject status="403" reason="Denied by policy"/>
      </taa:rule>
    </taa:rule-switch>
  </taa:routed>
</cpl>
```

Overview

Regular expressions can be used in conjunction with a number of VCS features such as alias transformations, zone transformations, CPL policy and ENUM. The VCS uses POSIX format regular expression syntax.

The table opposite provides a list of commonly used special characters in regular expression syntax. This is only a subset of the full range of expressions available. For a detailed description of regular expression syntax see the publication [Mastering Regular Expressions \[9\]](#).



For an example of regular expression usage, see the [CPL examples](#) section.

Common regular expressions

Character	Description	Example
.	Matches any single character.	
*	Matches 0 or more repetitions of the previous match.	. * will match against any sequence of characters.
+	Matches 1 or more repetitions of the previous match.	
\	Escapes a regular expression special character.	
\d	Matches any decimal digit, i.e. 0-9.	
[...]	Matches a set of characters. Each character in the set can be specified individually, or a range can be specified by giving the first character in the range followed by the - character and then the last character in the range. You can not use special characters within the [] - they will be taken literally.	[a-z] will match against any lower case alphabetical character. [a-zA-Z] will match against any alphabetical character. [0-9#*] will match against any single E.164 character - the E.164 character set is made up of the digits 0-9 plus the hash key (#) and the asterisk key (*).
(...)	Groups a set of matching characters together. Groups can then be referenced in order using the characters \1, \2, etc. as part of a replace string.	A regular expression can be constructed to transform a URI containing a user's full name to a URI based on their initials. The regular expression (.)*_(.)*(@example.com) would match against the user john_smith@example.com and with a replace string of \1\2\3 would transform it to js@example.com .
	Matches against one expression or an alternate expression.	.*@example.(net com) will match against any URI for the domain example.com or the domain example.net .
^	Signifies the start of a line. When used immediately after an opening brace, negates the character set inside the brace.	[^abc] matches any single character that is NOT one of a , b or c .
\$	Signifies the end of a line.	^d\d\d\$ will match any string that is exactly 3 digits long.
(?!...)	Negative lookahead. Defines a subexpression that <i>must not</i> be present in order for there to be a match.	(?!.*@example.com\$).* will match any string that does not end with @example.com .

Overview

The VCS makes use of pattern matching in a number of its features, namely [Allow Lists and Deny Lists](#), [pre-search transforms](#) and when configuring [search rules and zone transforms](#).

For each of these pattern matches, the VCS allows you to use a variable that it will replace with the current configuration value(s) before the pattern is checked.

These variables can be used as either or both of:

- all or part of the pattern that is being searched for
- all or part of the string that is replacing the pattern that was found.

The variables can be used in all types of patterns, i.e. **prefix**, **suffix**, **regex** and **exact**.

The table opposite shows the strings that are valid as variables, and the values they represent.



You can test whether a pattern will match a particular alias and be transformed in the expected way by using the [Check pattern](#) page (**Maintenance > Tools > Check pattern**).

Valid variable strings

String	Equals current value(s) returned by...	When used in a Pattern field	When used in a Replace field
%ip%	xConfiguration Ethernet 1 IP V4 Address xConfiguration Ethernet 1 IP V6 Address xConfiguration Ethernet 2 IP V4 Address xConfiguration Ethernet 2 IP V6 Address	Matches all IPv4 and IPv6 addresses currently configured on the VCS.	not applicable
%ip%v4%	xConfiguration Ethernet 1 IP V4 Address xConfiguration Ethernet 2 IP V4 Address	Matches the IPv4 addresses currently configured on the VCS for LAN 1 and LAN 2.	not applicable
%ip%v4 _ 1%	xConfiguration Ethernet 1 IP V4 Address	Matches all IPv4 address currently configured on the VCS for LAN 1.	Replaces the string with the LAN 1 IPv4 address.
%ip%v4 _ 2%	xConfiguration Ethernet 2 IP V4 Address	Matches all IPv4 address currently configured on the VCS for LAN 2.	Replaces the string with the LAN 2 IPv4 address.
%ip%v6%	xConfiguration Ethernet 1 IP V6 Address xConfiguration Ethernet 2 IP V6 Address	Matches the IPv6 addresses currently configured on the VCS for LAN 1 and LAN 2.	not applicable
%ip%v6 _ 1%	xConfiguration Ethernet 1 IP V6 Address	Matches the IPv6 address currently configured on the VCS for LAN 1.	Replaces the string with the LAN 1 IPv6 address.
%ip%v6 _ 2%	xConfiguration Ethernet 2 IP V6 Address	Matches the IPv6 address currently configured on the VCS for LAN 2.	Replaces the string with the LAN 2 IPv6 address.
%localdomains%	xConfiguration SIP Domains Domain 1 Name ... xConfiguration SIP Domains Domain 20 Name	Matches all the SIP domains currently configured on the VCS.	not applicable
%localdomain1% ... %localdomain20%	xConfiguration SIP Domains Domain 1 Name ... xConfiguration SIP Domains Domain 20 Name	Matches the specified SIP domain. Up to 20 SIP domains can be configured on the VCS, and they are identified by an index number between 1 and 20.	Replaces the string with the specified SIP domain.
%systemname%	xConfiguration SystemUnit Name	Matches the VCS's System Name.	Replaces the string with the VCS's System Name.

Overview

The VCS uses different IP ports and protocols for different services and functions, and many of these are configurable. The table below lists each of these services and functions. For each, it shows the default port(s) and protocol used and whether these ports are used for inbound or outbound communications. If the ports are configurable it shows the available range and how to configure them using the web interface or CLI.



The information in the table below shows all possible services and the generic defaults for each. The actual services and ports used on your system will vary depending on its configuration, the option keys installed and features that have been enabled. A specific list of all the IP ports in use on a particular VCS can be viewed via the **Maintenance > Tools > Port usage** pages. See the [Port usage](#) section for more information.



Two services or functions cannot share the same port and protocol; if you attempt to change an existing port or range and it conflicts with another service, you will get a warning message.

Ports

Service/function	Description	Default	Direction	Available range	Configurable via
SSH and FindMe replication for clusters	Used for encrypted command line administration. Also used to replicate FindMe data if the VCS is part of a cluster with FindMe enabled.	22 TCP	inbound	not configurable	
Telnet	Used for unencrypted command line administration.	23 TCP	inbound	not configurable	
HTTP	Used for unencrypted web administration.	80 TCP	inbound	not configurable	
NTP	Used for updating the system time (and important for H.235 security).	123 UDP	outbound	not configurable	
SNMP	Used for network management.	161 UDP	inbound	not configurable	
TMS Agent	Used for diagnostics.	389 TCP	inbound	not configurable	
HTTPS	Used for encrypted web administration. Also used to replicate FindMe data if the VCS is part of a cluster with FindMe enabled.	443 TCP	inbound	not configurable	
Reserved for future use		636	inbound	not configurable	
Gatekeeper discovery	Used for multicast gatekeeper discovery.	1718 UDP	inbound	not configurable	
H.323 registration Clustering	Listens for inbound H.323 UDP registrations. If the VCS is part of a cluster, this port is also used for inbound and outbound communication with peers, even if H.323 is disabled.	1719 UDP	inbound outbound	1024 - 65534	VCS configuration > Protocols > H.323 xConfiguration H323 Gatekeeper Registration UDP Port
H.323 call signaling	Listens for H.323 call signaling.	1720 TCP	inbound	1024 - 65534	VCS configuration > Protocols > H.323 xConfiguration H323 Gatekeeper CallSignaling TCP Port

Port reference

Ports

Service/function	Description	Default	Direction	Available range	Configurable via
Traversal server media demultiplexing RTP	Used on the VCS Expressway for demultiplexing RTP media.	2776 UDP	inbound outbound	1024 - 65534	VCS configuration > Expressway > Ports xConfiguration Traversal Server Media Demultiplexing RTP Port
Assent call signaling	Used on the VCS Expressway for Assent signaling.	2776 TCP	inbound	1024 - 65534	VCS configuration > Expressway > Ports xConfiguration Traversal Server H323 Assent CallSignaling Port
H.460.18 call signaling	Used on the VCS Expressway for H.460.18 signaling.	2777 TCP	inbound	1024 - 65534	VCS configuration > Expressway > Ports xConfiguration Traversal Server H323 H46018 CallSignaling Port
Traversal server media demultiplexing RTCP	Used on the VCS Expressway for demultiplexing RTCP media.	2777 UDP	inbound outbound	1024 - 65534	VCS configuration > Expressway > Ports xConfiguration Traversal Server Media Demultiplexing RTCP Port
TURN services	VCS Expressway listening port for TURN relay requests.	3478 UDP	inbound	1024 - 65534	VCS configuration > Expressway > TURN xConfiguration Traversal Server TURN Port
VCS database and TMS Agent (for clusters or Cisco TMS)	Encrypted administration connector to the VCS database. Used if the VCS is part of a cluster with FindMe or Device Provisioning enabled, or if the VCS is managed through Cisco TMS.	4444 TCP	inbound	not configurable	
SIP UDP	Listens for incoming SIP UDP calls.	5060 UDP	inbound outbound	1024 - 65534	VCS configuration > Protocols > SIP > Configuration xConfiguration SIP UDP Port
SIP TCP	Listens for incoming SIP TCP calls.	5060 TCP	inbound	1024 - 65534	VCS configuration > Protocols > SIP > Configuration xConfiguration SIP TCP Port
SIP TLS	Listens for incoming SIP TLS calls.	5061 TCP	inbound	1024 - 65534	VCS configuration > Protocols > SIP > Configuration xConfiguration SIP TLS Port
Traversal server zone H323 Port	The port on the VCS Expressway being used for H.323 firewall traversal from a particular traversal client.	6001 UDP, increments by 1 for each new zone	inbound	1024 - 65534	VCS configuration > Zones > Edit zone xConfiguration Zones Zone [1..1000] TraversalServer H323 Port
Traversal server zone SIP Port	The port on the VCS Expressway being used for SIP firewall traversal from a particular traversal client.	7001 TCP, increments by 1 for each new zone	inbound	1024 - 65534	VCS configuration > Zones > Edit zone xConfiguration Zones Zone [1..1000] TraversalServer SIP Port
TMS Agent	Used for Device Provisioning and FindMe.	8989 TCP	inbound	not configurable	

Ports

Service/function	Description	Default	Direction	Available range	Configurable via
DNS	Used for sending requests to DNS servers.	10000 - 10210 UDP	outbound	not configurable	
H.225 and H.245 call signaling port range	The range of ports to be used for call signalling after a call is established.	15000 - 19999 TCP	inbound outbound	1024 - 65534	VCS configuration > Protocols > H.323 xConfiguration H323 Gatekeeper CallSignaling PortRange Start xConfiguration H323 Gatekeeper CallSignaling PortRange End
SIP TCP outbound port range	The range of ports to be used by outbound TCP/TLS SIP connections to a remote SIP device.	25000 - 29999 TCP	outbound	1024 - 65534	VCS configuration > Protocols > SIP > Configuration xConfiguration SIP TCP Outbound Port Start xConfiguration SIP TCP Outbound Port End
Traversal media port range	For traversal calls (where the VCS takes the media as well as the signaling), the range of ports to be used for the media. Ports are allocated from this range in pairs, with the first port number of each pair being an even number. See Configuring the Traversal Subzone ports for more information.	50000 - 52399 UDP	outbound	1024 - 65533	VCS configuration > Local Zone > Traversal Subzone xConfiguration Traversal Media Port Start xConfiguration Traversal Media Port End
TURN relay media port range	The range of ports available for TURN media relay.	60000 - 61200 UDP	inbound outbound	1024 - 65534	VCS configuration > Expressway > TURN xConfiguration Traversal Server TURN Media Port Start xConfiguration Traversal Server TURN Media Port End
LDAP	Used for outbound connection to an LDAP server (if the VCS is configured to use an LDAP server for H.350 authentication).	uses a TCP source port from the ephemeral range			xConfiguration IP Ephemeral PortRange Start xConfiguration IP Ephemeral PortRange End
External manager	Used for outbound connection to an external manager e.g. Cisco TMS.	uses a TCP source port from the ephemeral range			
Third-party FindMe / User Policy server	Used for outbound connection to a third-party FindMe / User Policy server.	uses a TCP source port from the ephemeral range			
Remote logging	Used to send messages to the remote syslog server.	uses a UDP source port from the ephemeral range			
TMS Agent	Used to connect to another VCS or Cisco TMS for data replication.	uses a TCP source port from the ephemeral range			

Overview

This section gives examples of DNS configuration using Microsoft DNS Server and BIND 8 & 9.

These examples show how to set up an SRV record to handle H.323 URLs of the form *user@example.com*. These are handled by the system with the fully qualified domain name of *vcs.example.com* which is listening on port 1719, the default registration port.



It is assumed that both A and AAAA records already exist for *vcs.example.com*. If not, you will need to add one.

Verifying the SRV record

There are a range of tools available to investigate DNS records. One commonly found on Microsoft Windows and UNIX platforms is **nslookup**. Use this to verify that everything is working as expected.

For example:

- **nslookup -querytype=srv _h323ls._udp.example.com**
and check the output.

Microsoft DNS server

Using Microsoft DNS Server you can add the SRV record using either the command line or the MMC snap-in.

To use the command line, on the DNS server open a command window and enter:

- **dnscmd . /RecordAdd domain service_name SRV Priority Weight Port Target**
where:

domain	is the domain into which you wish to insert the record
service_name	is the name of the service you're adding
Priority	is the priority as defined by RFC 2782 [3]
Weight	is the weight as defined by RFC 2782 [3]
Port	is the port on which the system hosting the domain is listening
Target	is the FQDN of the system hosting the domain

For example:

- **dnscmd . /RecordAdd example.com _h323ls._udp SRV 1 0 1719 vcs.example.com**

BIND 8 & 9

BIND is a commonly used DNS server on UNIX and Linux systems. Configuration is based around two sets of text files: **named.conf** which describes which zones are represented by the server, and a selection of zone files which describe the detail of each zone.

BIND is sometimes run chrooted for increased security. This gives the program a new root directory, which means that the configuration files may not appear where you expect them to be. To see if this is the case on your system, run

- **ps aux | grep named**

This will give the command line that named (the BIND server) was invoked with. If there is a **-t** option, then the path following that is the new root directory and your files will be located relative to that root.

In **/etc/named.conf** look for a directory entry within the options section. This will give the directory in which the zone files are stored, possibly relative to a new root directory. In the appropriate zone section, a file entry will give the name of the file containing the zone details.



For more details of how to configure BIND servers and the DNS system in general see the publication [DNS and BIND \[6\]](#).

About the LDAP databases

The VCS can be configured to use a database on an LDAP Directory Server to store device authentication credential information (usernames, passwords, and other relevant information).

This section describes how to download the schemas that must be installed on the LDAP server, and how to install and configure two common types of LDAP servers, Microsoft Active Directory and OpenLDAP, for use with the VCS.

Downloading the H.350 schemas

The following ITU specifications describe the schemas which are required to be installed on the LDAP server:

H.350	Directory services architecture for multimedia conferencing - an LDAP schema to represent endpoints on the network.
H.350.1	Directory services architecture for H.323 - an LDAP schema to represent H.323 endpoints.
H.350.2	Directory services architecture for H.235 - an LDAP schema to represent H.235 elements.
H.350.4	Directory services architecture for SIP - an LDAP schema to represent SIP endpoints.

The schemas can be downloaded in **ldif** format from the web interface on the VCS. To do this:

1. Go to **VCS configuration > Authentication > Devices > LDAP Schemas**. You are presented with a list of downloadable schemas.
2. Click on the **Download** button next to each file to open it.
3. Use your browser's Save As command to store it on your file system.

Microsoft Active Directory

Prerequisites

These step-by-step instructions assume that Active Directory has already been installed. For details on installing Active Directory please consult your Windows documentation.

The following instructions are for Windows Server 2003 Enterprise Edition. If you are not using this version of Windows, your instructions may vary.

Installing the H.350 schemas

After you have [downloaded the H.350 schemas](#), install them as follows:

Open a command prompt and for each file execute the following command:

```
ldifde -i -c DC=X <ldap_base> -f filename.ldf
```

where:

<ldap_base> is the base DN for your Active Directory server.

Microsoft Active Directory

Adding H.350 objects

Create the organizational hierarchy

1. Open up the Active Directory **Users and Computers** MMC snap-in.
2. Under your BaseDN right-click and select **New Organizational Unit**.
3. Create an Organizational unit called **h350**.



It is good practice to keep the H.350 directory in its own organizational unit to separate out H.350 objects from other types of objects. This allows access controls to be setup which only allow the VCS read access to the BaseDN and therefore limit access to other sections of the directory.

Add the H.350 objects

1. Create an **ldif** file with the following contents:

```
# MeetingRoom1 endpoint
dn: commUniqueId=comm1,ou=h350,DC=X
objectClass: commObject
objectClass: h323Identity
objectClass: h235Identity
objectClass: SIPIdentity
commUniqueId: comm1
h323Identityh323-ID: MeetingRoom1
h323IdentitydialedDigits: 626262
h235IdentityEndpointID: meetingroom1
h235IdentityPassword: mypassword
SIPIdentityUserName: meetingroom1
SIPIdentityPassword: mypassword
SIPIdentitySIPURI: sip:MeetingRoom@X
```

2. Add the **ldif** file to the server using the command:

```
ldifde -i -c DC=X <ldap_base> -f filename.ldf
where:
```

<ldap_base> is the base DN of your Active Directory Server.

The example above will add a single endpoint with an H.323 ID alias of **MeetingRoom1**, an E.164 alias of **626262** and a SIP URI of **MeetingRoom@X**. The entry also has H.235 and SIP credentials of ID **meetingroom1** and password **mypassword** which are used during authentication.

H.323 registrations will look for the H.323 and H.235 attributes; SIP will look for the SIP attributes. Therefore if your endpoint is registering with just one protocol you do not need to include elements relating to the other.

Securing with TLS

To enable Active Directory to use TLS, you must request and install a certificate on the Active Directory server. The certificate must meet the following requirements:

- Be located in the Local Computer's Personal certificate store. This can be seen using the **Certificates** MMC snap-in.
- Have the private details on how to obtain a key associated for use with it stored locally. When viewing the certificate you should see a message saying "You have a private key that corresponds to this certificate".
- Have a private key that does not have strong private key protection enabled. This is an attribute that can be added to a key request.
- The Enhanced Key Usage extension includes the Server Authentication object identifier, again this forms part of the key request.
- Issued by a CA that both the domain controller and the client trust.
- Include the Active Directory fully qualified domain name of the domain controller in the common name in the subject field and/or the DNS entry in the subject alternative name extension.

To configure the VCS to use TLS on the connection to the LDAP server you must upload the CA's certificate as a trusted CA certificate. This can be done on the VCS by navigating to:

- **Maintenance > Security certificates**



The SIP URI in the **ldif** file must be prefixed by **sip:**.



For information about what happens when an alias is not in the LDAP database see the [Alias origin](#) section.

OpenLDAP

Prerequisites

These instructions assume that an OpenLDAP server has already been installed. For details on installing OpenLDAP see the documentation at <http://www.openldap.org>.

The following examples use a standard OpenLDAP installation on the Linux platform. For installations on other platforms the location of the OpenLDAP configuration files may be different. See the OpenLDAP installation documentation for details.

Installing the H.350 schemas

1. Copy the OpenLDAP files to the OpenLDAP schema directory:

```
/etc/openldap/schemas/commobject.ldif  
/etc/openldap/schemas/h323identity.ldif  
/etc/openldap/schemas/h235identity.ldif  
/etc/openldap/schemas/sipidentity.ldif
```

2. Edit `/etc/openldap/slapd.conf` to add the new schemas. You will need to add the following lines:

```
include /etc/openldap/schemas/commobject.ldif  
include /etc/openldap/schemas/h323identity.ldif  
include /etc/openldap/schemas/h235identity.ldif  
include /etc/openldap/schemas/sipidentity.ldif
```

The OpenLDAP daemon (`slapd`) must be restarted for the new schemas to take effect.

OpenLDAP

Adding H.350 objects

Create the organizational hierarchy

1. Create an `ldif` file with the following contents:

```
# This example creates a single
# organizational unit to contain the H.350
# objects
dn: ou=h350,dc=my-domain,dc=com
objectClass: organizationalUnit
ou: h350
```

2. Add the `ldif` file to the server using the command:

```
slapadd -l <ldif_file>
```

This organizational unit will form the BaseDN to which the VCS will issue searches. In this example the BaseDN will be: `ou=h350,dc=my-domain,dc=com`.



It is good practice to keep the H.350 directory in its own organizational unit to separate out H.350 objects from other types of objects. This allows access controls to be setup which only allow the VCS read access to the BaseDN and therefore limit access to other sections of the directory.

Add the H.350 objects

1. Create an `ldif` file with the following contents:

```
# MeetingRoom1 endpoint
dn: commUniqueId=comm1,ou=h350,dc=my-
domain,dc=com
objectClass: commObject
objectClass: h323Identity
objectClass: h235Identity
objectClass: SIPIdentity
commUniqueId: comm1
h323Identityh323-ID: MeetingRoom1
h323IdentitydialedDigits: 626262
h235IdentityEndpointID: meetingroom1
h235IdentityPassword: mypassword
SIPIdentityUserName: meetingroom1
SIPIdentityPassword: mypassword
SIPIdentitySIPURI: sip:MeetingRoom@domain.com
```

2. Add the `ldif` file to the server using the command:

```
slapadd -l <ldif_file>
```

The example above will add a single endpoint with an H.323 ID alias of **MeetingRoom1**, an E.164 alias of **626262** and a SIP URI of **MeetingRoom@domain.com**. The entry also has H.235 and SIP credentials of ID **meetingroom1** and password **mypassword** which are used during authentication.

H.323 registrations will look for the H.323 and H.235 attributes; SIP will look for the SIP attributes. Therefore if your endpoint is registering with just one protocol you do not need to include elements relating to the other.

Securing with TLS

The connection to the LDAP server can be encrypted by enabling Transport Level Security (TLS) on the connection. To do this you must create an X.509 certificate for the LDAP server to allow the VCS to verify the server's identity. After the certificate has been created you will need to install the following three files associated with the certificate onto the LDAP server:

- The certificate for the LDAP server.
- The private key for the LDAP server.
- The certificate of the Certificate Authority (CA) that was used to sign the LDAP server's certificate.

All three files should be in PEM file format.

The LDAP server must be configured to use the certificate. To do this:

- Edit `/etc/openldap/slapd.conf` and add the following three lines:

```
TLSCACertificateFile <path to CA certificate>
TLSCertificateFile <path to LDAP server certificate>
TLSCertificateKeyFile <path to LDAP private key>
```

The OpenLDAP daemon (**slapd**) must be restarted for the TLS settings to take effect.

To configure the VCS to use TLS on the connection to the LDAP server you must upload the CA's certificate as a trusted CA certificate. This can be done on the VCS by navigating to:

- **Maintenance > Security certificates**



The SIP URI in the `ldif` file must be prefixed by `sip:.`



For information about what happens when an alias is not in the LDAP database see the [Alias origin](#) section.

Overview

The **xConfiguration** group of commands are used to set and change individual items of configuration. Each command is made up of a main element followed by one or more sub-elements.

The following pages list all the **xConfiguration** commands currently available on the VCS.

To set a particular item of configuration, type the command as shown. The valid values for each command are indicated in the angle brackets following each command; these are explained opposite.

To obtain information about the existing configuration on the VCS:

- type **xConfiguration** to return all current configuration settings for the VCS.
- type **xConfiguration <element>** to return all current configuration for that particular element and all its sub-elements.
- type **xConfiguration <element> <sub-element>** to return all current configuration for that group of sub-elements.

To obtain information about using each of the **xConfiguration** commands:

- type **xConfiguration ?** to return a list of all elements available under the **xConfiguration** command.
- type **xConfiguration ??** to return a list of all elements available under the **xConfiguration** command, along with the valuespace, description and default values for each.
- type **xConfiguration <element> ?** to return all available sub-elements, along with the valuespace, description and default values for each.
- type **xConfiguration <element> <sub-element> ?** to return all available sub-elements, along with the valuespace, description and default values for each.

The valid value for this command is a string. The minimum and maximum number of characters is shown after the **S**.

When issuing this command, the string must be typed in double quotes.

The valid values for this command are one of the options shown within the angle brackets.

The valid value for this command is an integer. The minimum and maximum values are shown within the angle brackets.

Square brackets indicate that you can configure more than one of this particular item. Each item is assigned an index within the range shown.

IP Gateway: <S: 7..15>

Specifies the IPv4 gateway of the VCS. Note: you must restart the system for any changes to take effect.
Default: 127.0.0.1

Example: **xConfiguration IP Gateway: "192.168.127.0"**

IP QoS Mode: <None/DiffServ>

Specifies the type of QoS (Quality of Service) tags to apply to all signaling and media packets.

None: no specific QoS tagging is applied.

DiffServ: puts the specified Tag value in the TOS (Type Of Service) field of the IPv4 header or TC (Traffic Class) field of the IPv6 header.
Note: you must restart the system for any changes to take effect.

Default: None

Example: **xConfiguration IP QoS Mode: DiffServ**

IP QoS Value: <0..63>

The value to be stamped onto all signaling and media traffic routed through the VCS. Note: you must restart the system for any changes to take effect.

Example: **xConfiguration IP QoS Value: 16**

IP Route [1..50] Address: <S: 0, 39>

Specifies an IP address used in conjunction with the Prefix Length to determine the network to which the route applies.

Administration HTTP Mode: <On/Off>

Determines whether HTTP calls will be redirected to the HTTPS port.

On: calls will be redirected to HTTPS.

Off: no HTTP access will be available.

Note: you must restart the system for any changes to take effect.

Default: On

Example: **xConfiguration Administration HTTP Mode: On**

Administration HTTPS Mode: <On/Off>

Determines whether the VCS can be accessed via the web interface. This must be **On** to enable both web interface and TMS access.

Note: you must restart the system for any changes to take effect.

Default: On

Example: **xConfiguration Administration HTTPS Mode: On**

Administration HTTPS RequireClientCertificate: <On/Off>

Determines whether the VCS requires a valid client certificate from your web browser before setting up an HTTPS session. Note: this does not affect client verification of the VCS's server certificate.

Default: Off

Example: **xConfiguration Administration HTTPS RequireClientCertificate: On**

Administration SSH Mode: <On/Off>

Determines whether the VCS can be accessed via SSH and SCP.

Note: you must restart the system for any changes to take effect.

Default: On

Example: **xConfiguration Administration SSH Mode: On**

Administration Telnet Mode: <On/Off>

Determines whether the VCS can be accessed via Telnet.

Note: you must restart the system for any changes to take effect.

Default: Off

Example: **xConfiguration Administration Telnet Mode: Off**

Administration TimeOut: <0..10000>

Sets the number of minutes that an administration session (HTTPS, Telnet or SSH) may be inactive before the session is timed out. A value of **0** turns session time outs off.

Default: 0

Example: **xConfiguration Administration TimeOut: 0**

Command reference - xConfiguration

Alternates Cluster Name: <S: 0..128>

The fully qualified domain name used in SRV records that address this VCS cluster, for example "cluster1.example.com". The name can only contain letters, digits, hyphens and underscores. Warning: if you change the cluster name after any user accounts have been configured on this VCS, you may need to reconfigure your user accounts to use the new cluster name. Refer to the [Clustering and peers](#) section for more information.

Example: **xConfiguration Alternates Cluster Name: "Regional"**

Alternates ConfigurationMaster: <1..6>

Specifies which peer in this cluster is the master, from which configuration will be replicated to all other peers. A cluster consists of up to 6 peers, including the local VCS.

Example: **xConfiguration Alternates ConfigurationMaster: 1**

Alternates Peer [1..6] Address: <S: 0, 128>

Specifies the IP address of one of the peers in the cluster to which this VCS belongs. A cluster consists of up to 6 peers, including the local VCS.

Note: this must be a valid IPv4 or IPv6 address.

Example: **xConfiguration Alternates 1 Peer Address: "10.13.0.2"**

Applications ConferenceFactory Alias: <S:0..60>

The alias that will be dialed by the endpoints when the Multiway feature is activated. This must be pre-configured on all endpoints that may be used to initiate the Multiway feature.

Example: **xConfiguration Applications ConferenceFactory Alias: "multiway@example.com"**

Applications ConferenceFactory Mode: <On/Off>

The Mode option allows you to enable or disable the Conference Factory application.

Default: Off

Example: **xConfiguration Applications ConferenceFactory Mode: Off**

Applications ConferenceFactory Range End: <1..65535>

The last number of the range that replaces %% in the template used to generate a conference alias.

Default: 65535

Example: **xConfiguration Applications ConferenceFactory Range End: 30000**

Applications ConferenceFactory Range Start: <1..65535>

The first number of the range that replaces %% in the template used to generate a conference alias.

Default: 65535

Example: **xConfiguration Applications ConferenceFactory Range Start: 10000**

Applications ConferenceFactory Template: <S:0..60>

The alias that the VCS will tell the endpoint to dial in order to create a Multiway conference on the MCU.

Note: this alias must route to the MCU as a fully-qualified SIP alias

Example: **Applications ConferenceFactory Template: "563%%@example.com"**

Command reference - xConfiguration

Applications External Status [1..10] Filename: <S:0..255>

XML file containing status that is to be attached for an external application.

Example: **xConfiguration Applications External Status 1 Filename: "foo.xml"**

Applications External Status [1..10] Name: <S:0..64>

Descriptive name for the external application whose status is being referenced.

Example: **xConfiguration Applications External Status 1 Name: "foo"**

Applications OCS Relay Mode: <On/Off>

Enables or disables OCS relay support.

Default: Off

Example: **xConfiguration Applications OCS Relay Mode: Off**

Applications OCS Relay OCS Domain: <S:0..128>

The SIP domain in use on the Microsoft Office Communications Server. This must be selected from one of the SIP domains already configured on the VCS, and must be the same domain used by all FindMe names.

Example: **xConfiguration Applications OCS Relay OCS Domain: "example.com"**

Applications OCS Relay OCS Routing Prefix: <S:0..128>

Prefix applied to requests routed through the VCS proxy, from the OCS Relay. This is then used to route the requests onto to the appropriate Microsoft Office Communications Server (via Neighbor zone matches).

Default: ocs

Example: **xConfiguration Applications OCS Relay OCS Routing Prefix: "ocs"**

Applications Presence Server Mode: <On/Off>

Enables and disables the SIMPLE Presence Server. Note: SIP mode must also be enabled for the Presence Server to function.

Default: Off

Example: **xConfiguration Applications Presence Server Mode: On**

Applications Presence Server Publication ExpireDelta: <30..7200>

Specifies the maximum time (in seconds) within which a publisher must refresh its publication.

Default: 120

Example: **xConfiguration Applications Presence Server Publication ExpireDelta: 120**

Applications Presence Server Subscription ExpireDelta: <30..7200>

Specifies the maximum time (in seconds) within which a subscriber must refresh its subscription.

Default: 300

Example: **xConfiguration Applications Presence Server Subscription ExpireDelta: 300**

Command reference - xConfiguration

Applications Presence User Agent ExpireDelta: <1..65534>

Specifies the lifetime value (in seconds) the Presence User Agent will advertise in the PUBLISH messages it sends to the Presence Server. The Presence User Agent will refresh its PUBLISH messages at 75% of this value (to keep them active). The Presence Server may reduce this value in its responses.

Default: 3600

Example: **xConfiguration Applications Presence User Agent ExpireDelta: 3600**

Applications Presence User Agent Mode: <On/Off>

Enables and disables the SIMPLE Presence User Agent (PUA). The PUA provides presence information on behalf of registered endpoints. SIP mode must also be enabled for the PUA to function.

Default: Off

Example: **xConfiguration Applications Presence User Agent Mode: Off**

Applications Presence User Agent RetryDelta: <1..65534>

Specifies the time (in seconds) after which the Presence User Agent will attempt to resend a PUBLISH to the Presence Server. This will occur if the original attempt failed due to resource issues or other transitory errors.

Default: 5

Example: **xConfiguration Applications Presence User Agent RetryDelta: 5**

Authentication Credential [1..2500] Name: <S: 0, 128>

Defines the name for this entry in the local authentication database.

Example: **xConfiguration Authentication Credential 1 Name: "john smith"**

Authentication Credential [1..2500] Password: <S: 0, 215>

Defines the password for this entry in the local authentication database. The maximum plaintext length is 128 characters, which will then be encrypted.

Example: **xConfiguration Authentication Credential 1 Password: "password123"**

Authentication Database: <LocalDatabase/LDAPDatabase>

Selects between a local authentication database and a remote LDAP repository for the storage of password information for authentication.

Default: LocalDatabase

Example: **xConfiguration Authentication Database: LocalDatabase**

Authentication LDAP AliasOrigin: <LDAP/Endpoint/Combined>

Determines which aliases (i.e. from the **LDAP** repository or the **endpoint**) should be used to register the endpoint.

Combined: the endpoint will be registered both with the aliases which it has presented and with those configured in the LDAP repository.

Default: LDAP

Example: **xConfiguration Authentication LDAP AliasOrigin: LDAP**

Authentication LDAP BaseDN: <S: 0, 255>

Specifies the Distinguished Name to use when connecting to an LDAP server.

Example: **xConfiguration Authentication LDAP BaseDN: "dc=example,dc=company,dc=com"**

Command reference - xConfiguration

Authentication Mode: <On/Off>

Determines whether systems attempting to communicate with the VCS must authenticate with it first.

Off: incoming messages are not authenticated.

On: for H.323, any credentials in the message are checked against the authentication database. The message is allowed if the credentials match, or if there are no credentials in the message. For SIP, any messages originating from an endpoint in a local domain will be authenticated.

Default: Off

Example: **xConfiguration Authentication Mode: Off**

Authentication Password: <S: 0, 215>

The password used by the VCS when authenticating with another system. The maximum plaintext length is 128 characters, which is then encrypted. Note: this does not apply to traversal client zones.

Example: **xConfiguration Authentication Password: "password123"**

Authentication UserName: <S: 0, 128>

The username used by the VCS when authenticating with another system. Note: this does not apply to traversal client zones.

Example: **xConfiguration Authentication UserName: "VCS123"**

Bandwidth Default: <64..2048>

Sets the bandwidth (in kbps) to be used on calls managed by the VCS in cases where no bandwidth has been specified by the endpoint.

Default: 384

Example: **xConfiguration Bandwidth Default: 384**

Bandwidth Downspeed PerCall Mode: <On/Off>

Determines whether or not the VCS will attempt to downspeed a call if there is insufficient per-call bandwidth available to fulfill the request.

On: the VCS will attempt to place the call at a lower bandwidth.

Off: the call will be rejected.

Default: On

Example: **xConfiguration Bandwidth Downspeed PerCall Mode: On**

Bandwidth Downspeed Total Mode: <On/Off>

Determines whether or not the VCS will attempt to downspeed a call if there is insufficient total bandwidth available to fulfill the request.

On: the VCS will attempt to place the call at a lower bandwidth.

Off: the call will be rejected.

Default: On

Example: **xConfiguration Bandwidth Downspeed Total Mode: On**

Bandwidth Link [1..3000] Name: <S: 1, 50>

Assigns a name to this link.

Example: **xConfiguration Bandwidth Link 1 Name: "HQ to BranchOffice"**

Command reference - xConfiguration

Bandwidth Link [1..3000] Node1 Name: <S: 0, 50>

Specifies the first zone or subzone to which this link will be applied.

Example: **xConfiguration Bandwidth Link 1 Node1 Name: "HQ"**

Bandwidth Link [1..3000] Node2 Name: <S: 0, 50>

Specifies the second zone or subzone to which this link will be applied.

Example: **xConfiguration Bandwidth Link 1 Node2 Name: "BranchOffice"**

Bandwidth Link [1..3000] Pipe1 Name: <S: 0, 50>

Specifies the first pipe to be associated with this link.

Example: **xConfiguration Bandwidth Link 1 Pipe1 Name: "512Kb ASDL"**

Bandwidth Link [1..3000] Pipe2 Name: <S: 0, 50>

Specifies the second pipe to be associated with this link.

Example: **xConfiguration Bandwidth Link 1 Pipe2 Name: "2Gb Broadband"**

Bandwidth Pipe [1..1000] Bandwidth PerCall Limit: <1..100000000>

If this pipe has limited per-call bandwidth, sets the maximum amount of bandwidth (in kbps) available for any one call.

Default: 1920

Example: **xConfiguration Bandwidth Pipe 1 Bandwidth PerCall Limit: 256**

Bandwidth Pipe [1..1000] Bandwidth PerCall Mode: <Limited/Unlimited/NoBandwidth>

Determines whether or not this pipe is limiting the bandwidth of individual calls.

NoBandwidth: no bandwidth available. No calls can be made on this pipe.

Default: Unlimited

Example: **xConfiguration Bandwidth Pipe 1 Bandwidth PerCall Mode: Limited**

Bandwidth Pipe [1..1000] Bandwidth Total Limit: <1..100000000>

If this pipe has limited bandwidth, sets the maximum bandwidth (in kbps) available at any one time on the pipe.

Default: 500000

Example: **xConfiguration Bandwidth Pipe 1 Bandwidth Total Limit: 1024**

Bandwidth Pipe [1..1000] Bandwidth Total Mode: <Limited/Unlimited/NoBandwidth>

Determines whether or not this pipe is enforcing total bandwidth restrictions.

NoBandwidth: no bandwidth available. No calls can be made on this pipe.

Default: Unlimited

Example: **xConfiguration Bandwidth Pipe 1 Bandwidth Total Mode: Limited**

Command reference - xConfiguration

Bandwidth Pipe [1..1000] Name: <S: 1, 50>

Assigns a name to this pipe.

Example: **xConfiguration Bandwidth Pipe 1 Name: "512Kb ASDL"**

Call Loop Detection Mode: <On/Off>

Specifies whether the VCS will check for call loops.

Default: On

Example: **xConfiguration Call Loop Detection Mode: On**

Call Routed Mode: <Always/Optimal>

Specifies whether the VCS routes the signaling for calls.

Always: the VCS will always route the call signaling.

Optimal: if possible, the VCS will remove itself from the call signaling path, which may mean the call does not consume a call license.

Default: Always

Example: **xConfiguration Call Routed Mode: Always**

Call Services CallsToUnknownIPAddresses: <Off/Direct/Indirect>

Determines the way in which the VCS will attempt to call systems which are not registered with it or one of its neighbors.

Direct: allows an endpoint to make a call to an unknown IP address without the VCS querying any neighbors. The call setup would occur just as it would if the far end were registered directly to the local system.

Indirect: upon receiving a call to an unknown IP address, the VCS will query its neighbors for the remote address and if permitted will route the call through the neighbor.

Off: endpoints registered directly to the VCS may only call an IP address of a system also registered directly to that VCS.

Default: Indirect

Example: **xConfiguration Call Services CallsToUnknownIPAddresses: Indirect**

Call Services Fallback Alias: <S: 0, 60>

Specifies the alias to which incoming calls are placed for calls where the IP address or domain name of the VCS has been given but no callee alias has been specified.

Example: **xConfiguration Call Services Fallback Alias: "reception@example.com"**

Certification AdvancedAccountSecurity Mode: <On/Off>

Enables or disables advanced account security. Note: you must restart the system for any changes to take effect.

Default: Off

Example: **xConfiguration Certification AdvancedAccountSecurity Mode: On**

Core Dump Mode: <On/Off>

Controls whether application core dumps are enabled. Note: you must restart the system for any changes to take effect.

Default: Off

Example: **xConfiguration Core Dump Mode: Off**

Command reference - xConfiguration

Error Reports Mode: <On/Off>

Determines whether the VCS will automatically send details of application failures to a specified web service.

Default: Off

Example: **xConfiguration Error Reports Mode: Off**

Error Reports URL: <S: 0, 128>

The URL of the web service to which error reports are sent.

Default: <http://vcser.tandberg.com/submitapplicationerror/>

Example: **xConfiguration Error Reports URL: "http://vcser.tandberg.com/submitapplicationerror/"**

Ethernet [1..2] IP V4 Address: <S: 7..15>

Specifies the IPv4 address of the specified LAN port. Note: you must restart the system for any changes to take effect.

Example: **xConfiguration Ethernet 1 IP V4 Address: "192.168.10.10"**

Ethernet [1..2] IP V4 StaticNAT Address: <S:7..15>

If the VCS is operating in static NAT mode, this specifies the external public IPv4 address of that static NAT. Note: you must restart the system for any changes to take effect.

Example: **xConfiguration Ethernet 1 IP V4 StaticNAT Address: "64.22.64.85"**

Ethernet [1..2] IP V4 StaticNAT Mode: <On/Off>

Specifies whether the VCS is located behind a static NAT. Note: You must restart the system for any changes to take effect.

Default: Off

Example: **xConfiguration Ethernet 1 IP V4 StaticNAT Mode: On**

Ethernet [1..2] IP V4 SubnetMask: <S: 7..15>

Specifies the IPv4 subnet mask of the specified LAN port. Note: you must restart the system for any changes to take effect.

Example: **xConfiguration Ethernet 1 IP V4 SubnetMask: "255.255.255.0"**

Ethernet [1..2] IP V6 Address: <S: 0, 39>

Specifies the IPv6 address of the specified LAN port. Note: you must restart the system for any changes to take effect

Example: **xConfiguration Ethernet 1 IP V6 Address: "2001:db8::1428:57ab"**

Ethernet [1..2] Speed: <Auto/10half/10full/100half/100full/1000full>

Sets the speed of the Ethernet link from the specified LAN port. Use **Auto** to automatically configure the speed. Note: you must restart the system for any changes to take effect.

Default: Auto

Example: **xConfiguration Ethernet 1 Speed: Auto**

ExternalManager Address: <S: 0, 128>

Sets the IP address or Fully Qualified Domain Name (FQDN) of the external manager.

Example: **xConfiguration ExternalManager Address: "192.168.0.0"**

Command reference - xConfiguration

ExternalManager Path: <S: 0, 255>

Sets the URL of the external manager.

Default: tms/public/external/management/SystemManagementService.asmx

Example: **xConfiguration ExternalManager Path: "tms/public/external/management/SystemManagementService.asmx"**

ExternalManager Protocol: <HTTP/HTTPS>

The protocol used to connect to the external manager.

Default: HTTP

Example: **xConfiguration ExternalManager Protocol: HTTP**

ExternalManager Server Certificate Verification Mode: <On/Off>

Controls whether the certificate presented by the external manager is verified.

Default: On

Example: **xConfiguration ExternalManager Server Certificate Verification Mode: On**

H323 Gatekeeper AutoDiscovery Mode: <On/Off>

Determines whether or not the VCS responds to gatekeeper discovery requests from endpoints.

Default: On

Example: **xConfiguration H323 Gatekeeper AutoDiscovery Mode: On**

H323 Gatekeeper CallSignaling PortRange End: <1024..65534>

Specifies the upper port in the range to be used by calls once they are established.

Default: 19999

Example: **xConfiguration H323 Gatekeeper CallSignaling PortRange End: 19999**

H323 Gatekeeper CallSignaling PortRange Start: <1024..65534>

Specifies the lower port in the range to be used by calls once they are established.

Default: 15000

Example: **xConfiguration H323 Gatekeeper CallSignaling PortRange Start: 15000**

H323 Gatekeeper CallSignaling TCP Port: <1024..65534>

Specifies the port that listens for H.323 call signaling.

Default: 1720

Example: **xConfiguration H323 Gatekeeper CallSignaling TCP Port: 1720**

H323 Gatekeeper CallTimeToLive: <60..65534>

Specifies the interval (in seconds) at which the VCS polls the endpoints in a call to verify that they are still in the call.

Default: 120

Example: **xConfiguration H323 Gatekeeper CallTimeToLive: 120**

H323 Gatekeeper Registration ConflictMode: <Reject/Overwrite>

Determines how the system will behave if an endpoint attempts to register an alias currently registered from another IP address.

Reject: denies the registration.

Overwrite: deletes the original registration and replaces it with the new registration.

Default: Reject

Example: **xConfiguration H323 Gatekeeper Registration ConflictMode: Reject**

H323 Gatekeeper Registration UDP Port: <1024..65534>

Specifies the port to be used for H.323 UDP registrations.

Default: 1719

Example: **xConfiguration H323 Gatekeeper Registration UDP Port: 1719**

H323 Gatekeeper TimeToLive: <60..65534>

Specifies the interval (in seconds) at which an H.323 endpoint must re-register with the VCS in order to confirm that it is still functioning.

Default: 1800

Example: **xConfiguration H323 Gatekeeper TimeToLive: 1800**

H323 Gateway CallerId: <IncludePrefix/ExcludePrefix>

Specifies whether the prefix of the ISDN gateway is inserted into the caller's E.164 number presented on the destination endpoint. Including the prefix allows the recipient to directly return the call.

IncludePrefix: inserts the ISDN gateway's prefix into the source E.164 number.

ExcludePrefix: only displays the source E.164 number.

Default: ExcludePrefix

Example: **xConfiguration H323 Gateway CallerId: ExcludePrefix**

H323 Mode: <On/Off>

Determines whether or not the VCS will provide H.323 gatekeeper functionality.

Default: On

Example: **xConfiguration H323 Mode: On**

Interworking Encryption Mode: <Auto/Off>

Determines whether or not the VCS will allow encrypted calls between SIP and H.323 endpoints.

Off: interworked calls will never be encrypted.

Auto: interworked calls will be encrypted if the endpoints request it.

Default: Auto

Example: **xConfiguration Interworking Encryption Mode: Auto**

Command reference - xConfiguration

Interworking Mode: <On/Off/RegisteredOnly>

Determines whether or not the VCS will act as a gateway between SIP and H.323 calls.

Off: the VCS will not act as a SIP-H.323 gateway.

RegisteredOnly: the VCS will act as a SIP-H.323 gateway but only if at least one of the endpoints is locally registered.

On: the VCS will act as SIP-H.323 gateway regardless of whether the endpoints are locally registered.

Default: RegisteredOnly

Example: **xConfiguration Interworking Mode: RegisteredOnly**

IP DNS Domain Name: <S: 0, 128>

The name to be appended to an unqualified host name before querying the DNS server.

Used only when attempting to resolve unqualified domain names for NTP, LDAP, external manager and remote syslog servers.

Example: **xConfiguration IP DNS Domain Name: "example.com"**

IP DNS Hostname : <S: 0, 63>

Defines the DNS host name that this system is known by. Note that this is not the fully-qualified domain name, just the host label portion.

The name can only contain letters, digits, hyphens and underscores. The first character must be a letter and the last character must be a letter or a digit.

Example: **xConfiguration IP DNS Hostname: "localvcs"**

IP DNS Server [1..5] Address: <S: 0, 39>

Sets the IP address of up to 5 DNS servers to be used when resolving domain names.

Example: **xConfiguration IP DNS Server 1 Address: "192.168.12.0"**

IP Ephemeral PortRange End: <1024..65534>

Specifies the highest port in the range to be used for ephemeral outbound connections not otherwise constrained by VCS call processing.

Default: 49999

Example: **xConfiguration IP Ephemeral PortRange End: 49999**

IP Ephemeral PortRange Start: <1024..65534>

Specifies the lowest port in the range to be used for ephemeral outbound connections not otherwise constrained by VCS call processing.

Default: 40000

Example: **xConfiguration IP Ephemeral PortRange Start: 40000**

IP External Interface: <LAN1/LAN2>

Defines which LAN interface is externally facing.

Default: LAN1

Example: **xConfiguration IP External Interface: LAN1**

IP Gateway: <S: 7..15>

Specifies the IPv4 gateway of the VCS. Note: you must restart the system for any changes to take effect.

Default: 127.0.0.1

Example: **xConfiguration IP Gateway: "192.168.127.0"**

IP QoS Mode: <None/DiffServ>

Specifies the type of QoS (Quality of Service) tags to apply to all signaling and media packets.

None: no specific QoS tagging is applied.

DiffServ: puts the specified Tag value in the TOS (Type Of Service) field of the IPv4 header or TC (Traffic Class) field of the IPv6 header.

Note: you must restart the system for any changes to take effect.

Default: None

Example: **xConfiguration IP QoS Mode: DiffServ**

IP QoS Value: <0..63>

The value to be stamped onto all signaling and media traffic routed through the VCS. Note: you must restart the system for any changes to take effect.

Example: **xConfiguration IP QoS Value: 16**

IP Route [1..50] Address: <S: 0, 39>

Specifies an IP address used in conjunction with the Prefix Length to determine the network to which this route applies.

Example: **xConfiguration IP Route 1 Address: "128.168.0.0"**

IP Route [1..50] Gateway: <S: 0, 39>

Specifies the IP address of the Gateway for this route.

Example: **xConfiguration IP Route 1 Gateway: "192.168.0.0"**

IP Route [1..50] Interface: <Auto/LAN1/LAN2>

Specifies the LAN interface to use for this route. Auto: The VCS will select the most appropriate interface to use.

Default: Auto

Example: **xConfiguration IP Route 1 Interface: Auto**

IP Route [1..50] PrefixLength: <0..128>

Specifies the number of bits of the IP address which must match when determining the network to which this route applies.

Default: 32

Example: **xConfiguration IP Route 1 PrefixLength: 16**

IP V6 Gateway: <S: 0, 39>

Specifies the IPv6 gateway of the VCS. Note: you must restart the system for any changes to take effect.

Example: **xConfiguration IP V6 Gateway: "3dda:80bb:6::9:144"**

Command reference - xConfiguration

IPProtocol: <Both/IPv4/IPv6>

Selects whether the VCS is operating in IPv4, IPv6 or dual stack mode. Note: you must restart the system for any changes to take effect.

Default: IPv4

Example: **xConfiguration IPProtocol: IPv4**

LDAP Encryption: <Off/TLS>

Sets the encryption to use for the connection to the LDAP server.

Off: no encryption is used.

TLS: TLS encryption is used.

Default: Off

Example: **xConfiguration LDAP Encryption: Off**

LDAP Password: <S: 0, 122>

Sets the password to use when binding to the LDAP server. The maximum plaintext length is 60 characters, which is then encrypted.

Example: **xConfiguration LDAP Password: "password123"**

LDAP Server Address: <S: 0, 128>

Sets the IP address or Fully Qualified Domain Name (FQDN) of the LDAP server to use when making LDAP queries.

Example: **xConfiguration LDAP Server Address: "ldap.server.example.com"**

LDAP Server Port: <1..65534>

Sets the IP port of the LDAP server to use when making LDAP queries. Typically, non-secure connections use 389 and secure connections use 636.

Default: 389

Example: **xConfiguration LDAP Server Port: 389**

LDAP UserDN: <S: 0, 255>

Sets the user distinguished name to use when binding to the LDAP server.

Example: **xConfiguration LDAP UserDN: "user123"**

Log Level: <1..4>

Controls the granularity of Event Logging. **1** is the least verbose, **4** the most.

Note: this setting is not retrospective; it will determine which events are written to the Event Log from now onwards.

Default: 1

Example: **xConfiguration Log Level: 1**

Log Server Address: <S: 0, 128>

Specifies the IP address or Fully Qualified Domain Name (FQDN) of the remote syslog server to which the log will be written. This server must support the BSD syslog protocol. It cannot be another VCS.

Example: **xConfiguration Log Server Address: "syslog.server.example.com"**

Command reference - xConfiguration

Login Administrator Groups Group [1..30] Access: <None/ReadOnly/ReadWrite/Auditor>

Defines the access level for members of the specified administrator group.

None: no access allowed.

ReadOnly: configuration can only be viewed.

ReadWrite: configuration can be viewed and changed.

Auditor: allows access to the Event Log, Configuration Log and the Overview page only.

Default: ReadWrite

Example: **xConfiguration Login Administrator Groups Group 1 Access: ReadWrite**

Login Administrator Groups Group [1..30] Name: <S: 0..128>

Defines the name of an administrator group that determines which access rights members of the group have after they have been successfully authenticated to use the VCS.

Example: **xConfiguration Login Administrator Groups Group 1 Name: "VCS_Admin"**

Login Administrator Source: <Local/Remote>

Defines where administrator login credentials are authenticated before access is allowed to the VCS.

Remote: credentials are verified against an external credentials directory, for example Windows Active Directory.

Local: credentials are verified against a local database stored on the VCS.

Default: Local

Example: **xConfiguration Login Administrator Source: Local**

Login Remote LDAP BaseDN Accounts: <S: 0..255>

Sets the Distinguished Name to use as the base when searching for administrator and user accounts.

Example: **xConfiguration Login Remote LDAP BaseDN Accounts: "ou=useraccounts,dc=corporation,dc=int"**

Login Remote LDAP BaseDN Groups: <S: 0..255>

Sets the Distinguished Name to use as the base when searching for administrator and user groups.

Example: **xConfiguration Login Remote LDAP BaseDN Groups: "ou=groups,dc=corporation,dc=int"**

Login Remote LDAP CRLCheck: <None/Peer/All>

Specifies whether certificate revocation lists (CRLs) are checked when forming a TLS connection with the LDAP server. CRL data is uploaded to the VCS via the trusted CA certificate PEM file.

None: no CRL checking is performed.

Peer: only the CRL associated with the CA that issued the LDAP server's certificate is checked.

All: all CRLs in the trusted certificate chain of the CA that issued the LDAP server's certificate are checked.

Default: None.

Example: **xConfiguration Login Remote LDAP CRLCheck: Peer**

Command reference - xConfiguration

Login Remote LDAP DirectoryType: <ActiveDirectory>

Defines the type of LDAP directory that is being accessed.

ActiveDirectory: directory is Windows Active Directory.

Default: ActiveDirectory

Example: **xConfiguration Login Remote LDAP DirectoryType: ActiveDirectory**

Login Remote LDAP Encryption: <Off/TLS>

Sets the encryption to use for the connection to the LDAP server.

Off: no encryption is used.

TLS: TLS encryption is used.

Default: Off

Example: **xConfiguration Login Remote LDAP Encryption: Off**

Login Remote LDAP SASL: <None/DIGEST-MD5>

Sets the SASL (Simple Authentication and Security Layer) mechanism to use when binding to the LDAP server.

None: no mechanism is used.

DIGEST-MD5: The DIGEST-MD5 mechanism is used.

Default: DIGEST-MD5

Example: **xConfiguration Login Remote LDAP SASL: DIGEST-MD5**

Login Remote LDAP Server Address: <S: 0..128>

Sets the IP address or Fully Qualified Domain Name (FQDN) of the LDAP server to use when making LDAP queries.

Example: **xConfiguration Login Remote LDAP Server Address: "server.example.com"**

Login Remote LDAP Server FQDNResolution: <AddressRecord/SRVRecord>

Sets how the LDAP server address is resolved if specified as an FQDN.

AddressRecord: DNS A or AAAA record lookup.

SRVRecord: DNS SRV record lookup.

Default: AddressRecord

Example: **xConfiguration Login Remote LDAP Server FQDNResolution: AddressRecord**

Login Remote LDAP Server Port: <1..65534>

Sets the IP port of the LDAP server to use when making LDAP queries. Typically, non-secure connections use 389 and secure connections use 636.

Default: 389

Example: **xConfiguration Login Remote LDAP Server Port: 389**

Login Remote LDAP VCS BindDN: <S: 0..255>

Sets the user distinguished name to use when binding to the LDAP server.

Example: **xConfiguration Login Remote LDAP VCS BindDN: "VCSmanager"**

Command reference - xConfiguration

Login Remote LDAP VCS BindPassword: <S: 0..122>

Sets the password to use when binding to the LDAP server. The maximum plaintext length is 60 characters, which is then encrypted.

Example: **xConfiguration Login Remote LDAP VCS BindPassword: "password123"**

Login Remote LDAP VCS BindUsername: <S: 0..255>

Sets the username to use when binding to the LDAP server. Only applies if using SASL.

Example: **xConfiguration Login Remote LDAP VCS BindUsername: "VCSmanager"**

Login Remote Protocol: <LDAP>

The protocol used to connect to the external directory.

Default: LDAP

Example: **xConfiguration Login Remote Protocol: LDAP**

Login User Groups Group [1..15] Access: <None/ReadWrite>

Defines the access level for members of the specified user group.

None: no access allowed.

ReadWrite: configuration can be viewed and changed.

Default: ReadWrite

Example: **xConfiguration Login User Groups Group 1 Access: ReadWrite**

Login User Groups Group [1..15] Name: <S: 0..128>

Defines the name of a user group that determines which access rights members of the group have after they have been successfully authenticated to use the VCS.

Example: **xConfiguration Login User Groups Group 1 Name: "FindMeAccounts"**

Login User Source: <Local/Remote>

Defines where user login credentials are authenticated before access is allowed to the VCS.

Remote: credentials are verified against an external credentials directory, for example Windows Active Directory.

Local: credentials are verified against a local database stored on the VCS.

Default: Local

Example: **xConfiguration Login User Source: Local**

NTP Address: <S: 0, 128>

Sets the IP address or Fully Qualified Domain Name (FQDN) of the NTP server to be used when synchronizing system time.

Example: **xConfiguration NTP Address: "ntp.server.example.com"**

Option [1..64] Key: <S: 0, 90>

Specifies the option key of your software option. These are added to the VCS in order to add extra functionality, such as increasing the VCS's capacity. Contact your Cisco representative for further information.

Example: **xConfiguration Option 1 Key: "1X4757T5-1-60BAD5CD"**

Command reference - xConfiguration

Policy AdministratorPolicy Mode: <On/Off>

Enables and disables use of Call Policy.
Default: Off

Example: **xConfiguration Policy AdministratorPolicy Mode: Off**

Policy FindMe CallerID: <FindMeID/IncomingID>

Determines how the source of an incoming call is presented to the callee.
IncomingID: displays the address of the endpoint from which the call was placed.
FindMeID: displays the FindMe ID associated with the originating endpoint's address.
Default: IncomingID

Example: **xConfiguration Policy FindMe CallerId: FindMeID**

Policy FindMe Mode: <Off/On/ThirdPartyManager>

Configures how the FindMe application operates.
Off: disables FindMe.
On: enables FindMe.
ThirdPartyManager: uses an off-box, third-party FindMe manager.
Default: Off

Example: **xConfiguration Policy FindMe Mode: On**

Policy FindMe Server Address: <S: 0, 128>

Specifies the IP address or Fully Qualified Domain Name (FQDN) of the remote FindMe Manager.

Example: **xConfiguration Policy FindMe Server Address: "userpolicy.server.example.com"**

Policy FindMe Server Password: <S: 0, 82>

Specifies the password used by the VCS to log in and query the remote FindMe Manager. The maximum plaintext length is 30 characters, which will then be encrypted.

Example: **xConfiguration Policy FindMe Server Password: "password123"**

Policy FindMe Server Path: <S: 0, 255>

Specifies the URL of the remote FindMe Manager.
Default: service

Example: **xConfiguration Policy FindMe Server Path: "service"**

Policy FindMe Server Protocol: <HTTP/HTTPS>

Specifies the protocol used to connect to the remote FindMe Manager.
Default: HTTP

Example: **xConfiguration Policy FindMe Server Protocol: HTTP**

Command reference - xConfiguration

Policy FindMe Server UserName: <S: 0, 30>

Specifies the user name used by the VCS to log in and query the remote FindMe Manager.

Example: **xConfiguration Policy FindMe Server UserName: "user123"**

Registration AllowList [1..2500] Description: <S: 0..64>

A free-form description of the Allow List rule.

Example: **xConfiguration Registration AllowList [1..2500] Description: "Everybody at @example.com"**

Registration AllowList [1..2500] Pattern String: <S: 0, 60>

Specifies an entry to be added to the Allow List. If one of an endpoint's aliases matches one of the patterns in the Allow List, the registration will be permitted.

Example: **xConfiguration Registration AllowList 1 Pattern String: "john.smith@example.com"**

Registration AllowList [1..2500] Pattern Type: <Exact/Prefix/Suffix/Regex>

Specifies whether the entry in the Allow List is a prefix, suffix, regular expression, or must be matched exactly.

Exact: the string must match the alias character for character.

Prefix: the string must appear at the beginning of the alias.

Suffix: the string must appear at the end of the alias.

Regex: the string will be treated as a regular expression.

Default: Exact

Example: **xConfiguration Registration AllowList 1 Pattern Type: Exact**

Registration DenyList [1..2500] Description: <S: 0..64>

A free-form description of the Deny List rule.

Example: **xConfiguration Registration DenyList [1..2500] Description: "Anybody at @nuisance.com"**

Registration DenyList [1..2500] Pattern String: <S: 0, 60>

Specifies an entry to be added to the Deny List. If one of an endpoint's aliases matches one of the patterns in the Deny List, the registration will not be permitted.

Example: **xConfiguration Registration DenyList 1 Pattern String: "john.jones@example.com"**

Registration DenyList [1..2500] Pattern Type: <Exact/Prefix/Suffix/Regex>

Specifies whether the entry in the Deny List is a prefix, suffix, regular expression, or must be matched exactly.

Exact: the string must match the alias character for character.

Prefix: the string must appear at the beginning of the alias.

Suffix: the string must appear at the end of the alias.

Regex: the string will be treated as a regular expression.

Default: Exact

Example: **xConfiguration Registration DenyList 1 Pattern Type: Exact**

Registration RestrictionPolicy: <None/AllowList/DenyList>

Specifies the policy to be used when determining which endpoints may register with the system.

None: no restriction.

AllowList: only endpoints attempting to register with an alias listed on the Allow List may register.

DenyList: all endpoints, except those attempting to register with an alias listed on the Deny List, may register.

Default: None

Example: **xConfiguration Registration RestrictionPolicy: None**

ResourceUsage Warning Activation Level: <0..100>

Controls if and when the VCS will warn that it is approaching its maximum licensed capacity for calls or registrations. The number represents the percentage of the maximum that, when reached, will trigger a warning. 0: Warnings will never appear.

Default: 90

Example: **ResourceUsage Warning Activation Level: 90**

Services AdvancedMediaGateway Policy Mode: <On/Off>

Controls whether the policy rules are used to control access to the Advanced Media Gateway.

Default: Off

Example: **xConfiguration Services AdvancedMediaGateway Policy Mode: On**

Services AdvancedMediaGateway Policy Rules Rule [1..200] Action: <Allow/Deny>

The action to take if the source or destination alias of the call matches this policy rule.

Allow: the call can connect via the Advanced Media Gateway.

Deny: the call can connect but it will not use Advanced Media Gateway resources.

Default: Allow

Example: **xConfiguration Services AdvancedMediaGateway Policy Rules Rule 1 Action: Allow**

Services AdvancedMediaGateway Policy Rules Rule [1..200] Description: <S: 0..64>

A free-form description of the Advanced Media Gateway policy rule.

Example: **xConfiguration Services AdvancedMediaGateway Policy Rules Rule 1 Description: "Deny all calls to branch office"**

Services AdvancedMediaGateway Policy Rules Rule [1..200] Name: <S: 0..50>

Assigns a name to this Advanced Media Gateway policy rule.

Example: **xConfiguration Services AdvancedMediaGateway Policy Rules Rule 1 Name: "Deny branch calls"**

Services AdvancedMediaGateway Policy Rules Rule [1..200] Pattern String: <S: 0..60>

The pattern against which the alias is compared.

Example: **xConfiguration Services AdvancedMediaGateway Policy Rules Rule 1 Pattern String: ".branch@example.com"**

Services AdvancedMediaGateway Policy Rules Rule [1..200] Pattern Type: <Exact/Prefix/Suffix/Regex>

The way in which the pattern must match either the source or destination alias of the call.

Exact: the entire pattern string must exactly match the alias character for character.

Prefix: the pattern string must appear at the beginning of the alias.

Suffix: the pattern string must appear at the end of the alias.

Regex: the pattern string is treated as a regular expression.

Default: Exact

Example: **xConfiguration Services AdvancedMediaGateway Policy Rules Rule 1 Pattern Type: Suffix**

Services AdvancedMediaGateway Policy Rules Rule [1..200] Priority: <1..65534>

Determines the order in which the rules are applied. The rules with the highest priority (1, then 2, then 3 and so on) are applied first. If multiple rules have the same priority they are applied in configuration order.

Default: 100

Example: **xConfiguration Services AdvancedMediaGateway Policy Rules Rule 1 Priority: 50**

Services AdvancedMediaGateway Policy Rules Rule [1..200] State: <Enabled/Disabled>

Indicates if the policy rule is enabled or disabled. Disabled policy rules are ignored.

Default: Enabled

Example: **xConfiguration Services AdvancedMediaGateway Policy Rules Rule 1 State: Enabled**

Services AdvancedMediaGateway Zone Name: <S: 0..50>

Name of the zone that connects to the Advanced Media Gateway.

Example: **xConfiguration Services AdvancedMediaGateway Zone Name: "AM gateway zone"**

SIP Domains Domain [1..20] Name: <S: 0..128>

Specifies a domain for which this VCS is authoritative. The VCS will act as a SIP registrar and Presence Server for this domain, and will accept registration requests for any SIP endpoints attempting to register with an alias that includes this domain. The domain name can comprise multiple levels. Each level's name can only contain letters, digits and hyphens, with each level separated by a period (dot). A level name cannot start or end with a hyphen, and the final level name must start with a letter. An example valid domain name is "100.example-name.com".

Example: **xConfiguration SIP Domains Domain 1 Name: "100.example-name.com"**

SIP Mode: <On/Off>

Determines whether or not the VCS will provide SIP registrar and SIP proxy functionality. This mode must be enabled in order to use either the Presence Server or the Presence User Agent.

Default: On

Example: **xConfiguration SIP Mode: On**

SIP Registration ExpireDelta: <30..7200>

Specifies the period (in seconds) within which a SIP endpoint must re-register with the VCS to prevent its registration expiring.

Default: 60

Example: **xConfiguration SIP Registration ExpireDelta: 60**

Command reference - xConfiguration

SIP Registration Proxy Mode: <Off/ProxyToKnownOnly/ProxyToAny>

Specifies how proxied registrations should be handled.

Off: registration requests will not be proxied.

ProxyToKnownOnly: registration requests will be proxied to neighbors only.

ProxyToAny: registration requests will be proxied in accordance with the VCS's existing call processing rules.

Default: Off

Example: **xConfiguration SIP Registration Proxy Mode: Off**

SIP Require Duo Video Mode: <On/Off>

Controls whether the VCS will require the use of the `com.tandberg.sdp.duo.enable` extension for endpoints that support it.

Default: On

Example: **xConfiguration SIP Require Duo Video Mode: On**

SIP Require UDP BFCP Mode: <On/Off>

Controls whether the VCS will require the use of the `com.tandberg.udp.bfcp` extension for endpoints that support it.

Default: On

Example: **xConfiguration SIP Require UDP BFCP Mode: On**

SIP Routes Route [1..20] Address: <S:0..39>

Specifies the IP address of the next hop for this route, where matching SIP requests will be forwarded.

Note: this command is intended for developer use only.

Example: **xConfiguration SIP Routes Route 1 Address: "127.0.0.1"**

SIP Routes Route [1..20] Authenticated: <On/Off>

Whether to forward authenticated requests.

On: only forward requests along route if incoming message has been authenticated.

Off: always forward messages that match this route.

Default: Off

Note: this command is intended for developer use only.

Example: **xConfiguration SIP Routes Route 1 Authenticated: On**

SIP Routes Route [1..20] Header Name: <S:0..64>

Name of SIP header field to match (e.g. Event).

Note: this command is intended for developer use only.

Example: **xConfiguration SIP Routes Route 1 Header Name: "Event"**

Command reference - xConfiguration

SIP Routes Route [1..20] Header Pattern: <S:0..128>

Regular expression to match against the specified SIP header field.

Note: this command is intended for developer use only.

Example: **xConfiguration SIP Routes Route 1 Header Pattern: "(my-event-package)(.*)"**

SIP Routes Route [1..20] Method: <S:0..64>

SIP method to match to select this route (e.g. INVITE, SUBSCRIBE).

Note: this command is intended for developer use only.

Example: **xConfiguration SIP Routes Route 1 Method: "SUBSCRIBE"**

SIP Routes Route [1..20] Port: <1..65534>

Specifies the port on the next hop for this route to which matching SIP requests will be routed.

Default: 5060

Note: this command is intended for developer use only.

Example: **xConfiguration SIP Routes Route 1 Port: 22400**

SIP Routes Route [1..20] Request Line Pattern: <S:0..128>

Regular expression to match against the SIP request line.

Note: this command is intended for developer use only.

Example: **xConfiguration SIP Routes Route 1 Request Line Pattern: ".*@(%localdomains%|ip%)"**

SIP Routes Route [1..20] Tag: <S:0..64>

Tag value specified by external applications to identify routes that they create.

Note: this command is intended for developer use only.

Example: **xConfiguration SIP Routes Route 1 Tag: "Tag1"**

SIP Routes Route [1..20] Transport: <UDP/TCP/TLS>

Determines which transport type will be used for SIP messages forwarded along this route.

Default: TCP

Note: this command is intended for developer use only.

Example: **xConfiguration SIP Routes Route 1 Transport: TCP**

SIP Session Refresh Minimum: <90..7200>

The minimum value the VCS will negotiate for the session refresh interval for SIP calls. For further information refer to the definition of Min-SE header in [RFC 4028](#).

Default: 500

Example: **xConfiguration SIP Session Refresh Minimum: 500**

SIP Session Refresh Value: <90..7200>

The maximum time allowed between session refresh requests for SIP calls. For further information refer to the definition of Session-Expires in [RFC 4028](#).

Default: 1800

Example: **xConfiguration SIP Session Refresh Value: 1800**

SIP TCP Mode: <On/Off>

Determines whether incoming and outgoing SIP calls using the TCP protocol will be allowed.

Default: On

Example: **xConfiguration SIP TCP Mode: On**

SIP TCP Outbound Port End: <1024..65534>

Specifies the upper port in the range to be used by outbound TCP/TLS SIP connections.

Default: 29999

Example: **xConfiguration SIP TCP Outbound Port End: 29999**

SIP TCP Outbound Port Start: <1024..65534>

Specifies the lower port in the range to be used by outbound TCP/TLS SIP connections.

Default: 25000

Example: **xConfiguration SIP TCP Outbound Port Start: 25000**

SIP TCP Port: <1024..65534>

Specifies the listening port for incoming SIP TCP calls.

Default: 5060

Example: **xConfiguration SIP TCP Port: 5060**

SIP TLS Mode: <On/Off>

Determines whether incoming and outgoing SIP calls using the TLS protocol will be allowed.

Default: On

Example: **xConfiguration SIP TLS Mode: On**

SIP TLS Port: <1024..65534>

Specifies the listening port for incoming SIP TLS calls.

Default: 5061

Example: **xConfiguration SIP TLS Port: 5061**

SIP UDP Mode: <On/Off>

Determines whether incoming and outgoing SIP calls using the UDP protocol will be allowed.

Default: On

Example: **xConfiguration SIP UDP Mode: On**

Command reference - xConfiguration

SIP UDP Port: <1024..65534>

Specifies the listening port for incoming SIP UDP calls.
Default: 5060

Example: **xConfiguration SIP UDP Port: 5060**

SNMP CommunityName: <S: 0, 16>

Sets the VCS's SNMP community name.
Default: public

Example: **xConfiguration SNMP CommunityName: "public"**

SNMP Mode: <On/Off>

Enables or disables SNMP support.
Default: Off

Example: **xConfiguration SNMP Mode: On**

SNMP SystemContact: <S: 0, 70>

Specifies the name of the person who can be contacted regarding issues with the VCS.

Example: **xConfiguration SNMP SystemContact: "John Smith"**

SNMP SystemLocation: <S: 0, 70>

Specifies the physical location of the VCS.

Example: **xConfiguration SNMP SystemLocation: "Server Room 128"**

SystemUnit AdminAccount [1..15] Access: <AccountDisabled/ReadOnly/ReadWrite/Auditor >

Defines the access level of an administrator user who can login to the VCS web interface.

AccountDisabled: no access allowed.

ReadOnly: configuration can only be viewed.

ReadWrite: configuration can be viewed and changed.

Auditor: allows access to the Event Log, Configuration Log and the Overview page only.

Default: ReadWrite

Example: **xConfiguration SystemUnit AdminAccount 1 Access: ReadOnly**

SystemUnit AdminAccount [1..15] Name: <S:0..25>

Defines the name of an administrator user who can login to the VCS web interface.

Example: **xConfiguration SystemUnit AdminAccount 1 Name: "guest"**

SystemUnit AdminAccount [1..15] Password: <S:0..65>

Defines the password of an administrator user who can login to the VCS web interface. The maximum plaintext length is 16 characters, which will then be encrypted.

Example: **xConfiguration SystemUnit AdminAccount 1 Password: "password123"**

Command reference - xConfiguration

SystemUnit Maintenance Mode: <On/Off>

Sets the VCS into maintenance mode. New calls and registrations are disallowed and existing registrations are allowed to expire.

Default: Off

Example: **xConfiguration SystemUnit Maintenance Mode: Off**

SystemUnit Name: <S:, 0, 50>

Defines the name of the VCS. Choose a name that uniquely identifies the system.

Example: **xConfiguration SystemUnit Name: "Oslo HQ VCS"**

SystemUnit Password: <S: 0, 65>

Defines the password for the default 'admin' account. This account is used to log in to the VCS via Telnet, HTTP(S), SSH, SCP, and on the serial port. The maximum plaintext length is 16 characters, which will then be encrypted.

Example: **xConfiguration SystemUnit Password: "password123"**

SystemUnit StrictPassword Enforce: <On/Off>

Determines whether or not administrator passwords must meet a certain level of complexity before they are accepted.

Default: Off

Example: **SystemUnit StrictPassword Enforce: Off**

TimeZone Name: <S: 0, 64>

Sets the local time zone of the VCS. Time zone names follow the POSIX naming convention e.g. Europe/London or America/New_York.

Default: GMT

Example: **xConfiguration TimeZone Name: "GMT"**

Transform [1..100] Description: <S: 0..64>

A free-form description of the transform.

Example: **xConfiguration Transform [1..100] Description: "Change example.net to example.com"**

Transform [1..100] Pattern Behavior: <Strip/Replace>

How the alias is modified.

Strip: removes the matching prefix or suffix from the alias.

Replace: substitutes the matching part of the alias with the text in replace string.

AddPrefix: prepends the replace string to the alias.

AddSuffix: appends the replace string to the alias.

Default: Strip

Example: **xConfiguration Transform 1 Pattern Behavior: Replace**

Transform [1..100] Pattern Replace: <S: 0, 60>

The text string to use in conjunction with the selected Pattern behavior.

Example: **xConfiguration Transform 1 Pattern Replace: "example.com"**

Transform [1..100] Pattern String: <S: 0, 60>

The pattern against which the alias is compared.

Example: **xConfiguration Transform 1 Pattern String: "example.net"**

Transform [1..100] Pattern Type: <Exact/Prefix/Suffix/Regex>

How the pattern string must match the alias for the transform to be applied.

Exact: the entire string must exactly match the alias character for character.

Prefix: the string must appear at the beginning of the alias.

Suffix: the string must appear at the end of the alias.

Regex: the string is treated as a regular expression.

Default: Prefix

Example: **xConfiguration Transform 1 Pattern Type: Suffix**

Transform [1..100] Priority: <1..65534>

Assigns a priority to the specified transform. Transforms are compared with incoming aliases in order of priority, and the priority must be unique for each transform.

Default: 1

Example: **xConfiguration Transform 1 Priority: 10**

Transform [1..100] State: <Enabled/Disabled>

Indicates if the transform is enabled or disabled. Disabled transforms are ignored.

Example: **xConfiguration Transform 1 State: Enabled**

Traversal Media Port End: <1025..65533>

For traversal calls (i.e. where the VCS is taking the media as well as the signaling), specifies the upper port in the range to be used for the media. Ports are allocated from this range in pairs, the first of each being even. Therefore the range must end with an odd number.

Default: 52399

Example: **xConfiguration Traversal Media Port End: 52399**

Traversal Media Port Start: <1024..65532>

For traversal calls (i.e. where the VCS is taking the media as well as the signaling), specifies the lower port in the range to be used for the media. Ports are allocated from this range in pairs, the first of each being even. Therefore the range must start with an even number.

Default: 50000

Example: **xConfiguration Traversal Media Port Start: 50000**

Traversal Server H323 Assent CallSignaling Port: <1024..65534>

Specifies the port on the VCS to be used for Assent signaling.

Default: 2776

Example: **xConfiguration Traversal Server H323 Assent CallSignaling Port: 2777**

Command reference - xConfiguration

Traversal Server H323 H46018 CallSignaling Port: <1024..65534>

Specifies the port on the VCS to be used for H460.18 signaling.
Default: 2777

Example: **Traversal Server H323 H46018 CallSignaling Port: 2777**

Traversal Server Media Demultiplexing RTCP Port: <1024..65534>

Specifies the port on the VCS to be used for demultiplexing RTCP media. Note: You must restart the system for any changes to take effect.
Default: 2777

Example: **xConfiguration Traversal Server Media Demultiplexing RTCP Port: 2777**

Traversal Server Media Demultiplexing RTP Port: <1024..65534>

Specifies the port on the VCS to be used for demultiplexing RTP media. Note: You must restart the system for any changes to take effect.
Default: 2776

Example: **xConfiguration Traversal Server Media Demultiplexing RTP Port: 2776**

Traversal Server TURN Authentication Realm: <S: 1..128>

The realm sent by the server in its authentication challenges.
Default: TANDBERG

Example: **xConfiguration Traversal Server TURN Authentication Realm: "TANDBERG"**

Traversal Server TURN Media Port End: <1024..65534>

The upper port in the range used for TURN relays.
Default: 61399

Example: **xConfiguration Traversal Server TURN Media Port End: 61399**

Traversal Server TURN Media Port Start: <1024..65534>

The lower port in the range used for TURN relays.
Default: 60000

Example: **xConfiguration Traversal Server TURN Media Port Start: 60000**

Traversal Server TURN Mode: <On/Off>

Determines whether the VCS offers TURN services to traversal clients.
Default: Off

Example: **xConfiguration Traversal Server TURN Mode: Off**

Traversal Server TURN Port: <1024..65534>

The listening port for TURN requests.
Default: 3478

Example: **xConfiguration Traversal Server TURN Port: 3478**

Command reference - xConfiguration

Zones LocalZone DefaultSubZone Bandwidth PerCall Inter Limit: <1..100000000>

Specifies the bandwidth limit (in kbps) for any one call to or from an endpoint in the Default Subzone (applies only if the mode is set to **Limited**).
Default: 1920

Example: **xConfiguration Zones LocalZone DefaultSubZone Bandwidth PerCall Inter Limit: 1920**

Zones LocalZone DefaultSubZone Bandwidth PerCall Inter Mode: <Limited/Unlimited/NoBandwidth>

Determines whether there is a limit on the bandwidth for any one call to or from an endpoint in the Default Subzone.

NoBandwidth: no bandwidth available. No calls can be made to or from the Default Subzone.

Default: Unlimited

Example: **xConfiguration Zones LocalZone DefaultSubZone Bandwidth PerCall Inter Mode: Limited**

Zones LocalZone DefaultSubZone Bandwidth PerCall Intra Limit: <1..100000000>

Specifies the bandwidth limit (in kbps) for any one call between two endpoints within the Default Subzone (applies only if the mode is set to **Limited**).

Default: 1920

Example: **xConfiguration Zones LocalZone DefaultSubZone Bandwidth PerCall Intra Limit: 1920**

Zones LocalZone DefaultSubZone Bandwidth PerCall Intra Mode: <Limited/Unlimited/NoBandwidth>

Determines whether there is a limit on the bandwidth for any one call between two endpoints within the Default Subzone.

NoBandwidth: no bandwidth available. No calls can be made within the Default Subzone.

Default: Unlimited

Example: **xConfiguration Zones LocalZone DefaultSubZone Bandwidth PerCall Intra Mode: Limited**

Zones LocalZone DefaultSubZone Bandwidth Total Limit: <1..100000000>

Sets the total bandwidth limit (in kbps) of the Default Subzone (applies only if Mode is set to Limited).

Default: 500000

Example: **xConfiguration Zones LocalZone DefaultSubZone Bandwidth Total Limit: 500000**

Zones LocalZone DefaultSubZone Bandwidth Total Mode: <Limited/Unlimited/NoBandwidth>

Determines whether the Default Subzone has a limit on the total bandwidth being used by its endpoints at any one time.

NoBandwidth: no bandwidth available. No calls can be made to, from, or within the Default Subzone.

Default: Unlimited

Example: **xConfiguration Zones LocalZone DefaultSubZone Bandwidth Total Mode: Limited**

Zones LocalZone DefaultSubZone Registrations: <Allow/Deny>

Controls whether registrations assigned to the Default Subzone are accepted.

Default: Allow

Example: **xConfiguration Zones LocalZone DefaultSubZone Registrations: Allow**

Command reference - xConfiguration

Zones LocalZone SubZones MembershipRules Rule [1..3000] Description: <S: 0..64>

A free-form description of the membership rule.

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 Description: "Office-based staff"**

Zones LocalZone SubZones MembershipRules Rule [1..3000] Name: <S: 0..50>

Assigns a name to this membership rule.

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 Name: "Office Workers"**

Zones LocalZone SubZones MembershipRules Rule [1..3000] Pattern String: <S: 0..60>

Specifies the pattern against which the alias is compared.

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 Pattern String: "@example.com"**

Zones LocalZone SubZones MembershipRules Rule [1..3000] Pattern Type: <Exact/Prefix/Suffix/Regex>

The way in which the pattern must match the alias.

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 Pattern Type: Suffix**

Zones LocalZone SubZones MembershipRules Rule [1..3000] Priority: <1..65534>

Determines the order in which the rules are applied (and thus to which subzone the endpoint is assigned) if an endpoint's address satisfies multiple rules. The rules with the highest priority (1, then 2, then 3 and so on) are applied first. If multiple Subnet rules have the same priority the rule with the largest prefix length is applied first. Alias Pattern Match rules at the same priority are searched in configuration order.

Default: 100

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 Priority: 100**

Zones LocalZone SubZones MembershipRules Rule [1..3000] State: <Enabled/Disabled>

Indicates if the membership rule is enabled or disabled. Disabled membership rules are ignored.

Default: Enabled

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 State: Enabled**

Zones LocalZone SubZones MembershipRules Rule [1..3000] SubZoneName: <S: 0..50>

The subzone to which an endpoint is assigned if its address satisfies this rule.

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 SubZoneName: "Branch Office"**

Zones LocalZone SubZones MembershipRules Rule [1..3000] Subnet Address: <S: 0..39>

Specifies an IP address used (in conjunction with the prefix length) to identify this subnet.

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 Subnet Address: "192.168.0.0"**

Zones LocalZone SubZones MembershipRules Rule [1..3000] Subnet PrefixLength: <1..128>

The number of bits of the subnet address which must match for an IP address to belong in this subnet.

Default: 32

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 Subnet PrefixLength: 32**

Command reference - xConfiguration

Zones LocalZone SubZones MembershipRules Rule [1..3000] Type: <Subnet/AliasPatternMatch>

The type of address that applies to this rule.

Subnet: assigns the device if its IP address falls within the configured IP address subnet.

AliasPatternMatch: assigns the device if its alias matches the configured pattern.

Example: **xConfiguration Zones LocalZone SubZones MembershipRules Rule 1 Type: Subnet**

Zones LocalZone SubZones SubZone [1..1000] Bandwidth PerCall Inter Limit: <1..100000000>

Specifies the bandwidth limit (in kbps) on any one call to or from an endpoint in this subzone (applies only if Mode is set to **Limited**).

Default: 1920

Example: **xConfiguration Zones LocalZone SubZones SubZone 1 Bandwidth PerCall Inter Limit: 1920**

Zones LocalZone SubZones SubZone [1..1000] Bandwidth PerCall Inter Mode: <Limited/Unlimited/NoBandwidth>

Determines whether there is a limit on the bandwidth for any one call to or from an endpoint in this subzone.

NoBandwidth: no bandwidth available. No calls can be made to or from this subzone.

Default: Unlimited

Example: **xConfiguration Zones LocalZone SubZones SubZone 1 Bandwidth PerCall Inter Mode: Limited**

Zones LocalZone SubZones SubZone [1..1000] Bandwidth PerCall Intra Limit: <1..100000000>

Specifies the bandwidth limit (in kbps) for any one call between two endpoints within this subzone (applies only if the mode is set to **Limited**).

Default: 1920

Example: **Zones LocalZone SubZones SubZone 1 Bandwidth PerCall Intra Limit: 1920**

Zones LocalZone SubZones SubZone [1..1000] Bandwidth PerCall Intra Mode: <Limited/Unlimited/NoBandwidth>

Determines whether there is a limit on the bandwidth for any one call between two endpoints within this subzone.

NoBandwidth: no bandwidth available. No calls can be made within this subzone.

Default: Unlimited

Example: **xConfiguration Zones LocalZone SubZones SubZone 1 Bandwidth PerCall Intra Mode: Limited**

Zones LocalZone SubZones SubZone [1..1000] Bandwidth Total Limit: <1..100000000>

Sets the total bandwidth limit (in kbps) of this subzone (applies only if the mode is set to **Limited**).

Default: 500000

Example: **xConfiguration Zones LocalZone SubZones SubZone 1 Bandwidth Total Limit: 500000**

Zones LocalZone SubZones SubZone [1..1000] Bandwidth Total Mode: <Limited/Unlimited/NoBandwidth>

Determines whether this subzone has a limit on the total bandwidth of calls being used by its endpoints at any one time.

NoBandwidth: no bandwidth available. No calls can be made to, from, or within this subzone.

Default: Unlimited

Example: **xConfiguration Zones LocalZone SubZones SubZone 1 Bandwidth Total Mode: Limited**

Command reference - xConfiguration

Zones LocalZone SubZones SubZone [1..1000] Name: <S: 0, 50>

Assigns a name to this subzone.

Example: **xConfiguration Zones LocalZone SubZones SubZone 1 Name: "BranchOffice"**

Zones LocalZone SubZones SubZone [1..1000] Registrations: <Allow/Deny>

Controls whether registrations assigned to this subzone are accepted.

Default: Allow

Example: **xConfiguration Zones LocalZone SubZones SubZone 1 Registrations: Allow**

Zones LocalZone Traversal H323 Assent Mode: <On/Off>

Determines whether or not H.323 calls using Assent mode for firewall traversal will be allowed. Applies to traversal-enabled endpoints registered directly with the VCS.

Default: On

Example: **xConfiguration Zones LocalZone Traversal H323 Assent Mode: On**

Zones LocalZone Traversal H323 H46018 Mode: <On/Off>

Determines whether or not H.323 calls using H460.18 mode for firewall traversal will be allowed. Applies to traversal-enabled endpoints registered directly with the VCS.

Default: On

Example: **xConfiguration Zones LocalZone Traversal H323 H46018 Mode: On**

Zones LocalZone Traversal H323 H46019 Demultiplexing Mode: <On/Off>

Determines whether the VCS will operate in Demultiplexing mode for calls from traversal-enabled endpoints registered directly with it.

On: allows use of the same two ports for all calls.

Off: Each call will use a separate pair of ports for media.

Default: Off

Example: **xConfiguration Zones LocalZone Traversal H323 H46019 Demultiplexing Mode: Off**

Zones LocalZone Traversal H323 Preference: <Assent/H46018>

If an endpoint that is registered directly with the VCS supports both Assent and H460.18 protocols, this setting determines which the VCS uses.

Default: Assent

Example: **xConfiguration Zones LocalZone Traversal H323 Preference: Assent**

Zones LocalZone Traversal H323 TCPProbe KeepAliveInterval: <1..65534>

Sets the interval (in seconds) with which a traversal-enabled endpoint registered directly with the VCS will send a TCP probe to the VCS once a call is established, in order to keep the firewall's NAT bindings open.

Default: 20

Example: **xConfiguration Zones LocalZone Traversal H323 TCPProbe KeepAliveInterval: 20**

Command reference - xConfiguration

Zones LocalZone Traversal H323 TCPProbe RetryCount: <1..65534>

Sets the number of times traversal-enabled endpoints registered directly with the VCS will attempt to send a TCP probe to the VCS.
Default: 5

Example: **xConfiguration Zones LocalZone Traversal H323 TCPProbe RetryCount: 5**

Zones LocalZone Traversal H323 TCPProbe RetryInterval: <1..65534>

Sets the frequency (in seconds) with which traversal-enabled endpoints registered directly with the VCS will send a TCP probe to the VCS.
Default: 2

Example: **xConfiguration Zones LocalZone Traversal H323 TCPProbe RetryInterval: 2**

Zones LocalZone Traversal H323 UDPProbe KeepAliveInterval: <1..65534>

Sets the interval (in seconds) with which a traversal-enabled endpoint registered directly with the VCS will send a UDP probe to the VCS once a call is established, in order to keep the firewall's NAT bindings open.
Default: 20

Example: **xConfiguration Zones LocalZone Traversal H323 UDPProbe KeepAliveInterval: 20**

Zones LocalZone Traversal H323 UDPProbe RetryCount: <1..65534>

Sets the number of times traversal-enabled endpoints registered directly with the VCS will attempt to send a UDP probe to the VCS.
Default: 5

Example: **xConfiguration Zones LocalZone Traversal H323 UDPProbe RetryCount: 5**

Zones LocalZone Traversal H323 UDPProbe RetryInterval: <1..65534>

Sets the frequency (in seconds) with which traversal-enabled endpoints registered directly with the VCS will send a UDP probe to the VCS.
Default: 2

Example: **xConfiguration Zones LocalZone Traversal H323 UDPProbe RetryInterval: 2**

Zones LocalZone TraversalSubZone Bandwidth PerCall Limit: <1..100000000>

Specifies the bandwidth limit (in kbps) applied to any one traversal call being handled by the VCS (applies only if the mode is set to **Limited**).
Default: 1920

Example: **xConfiguration Zones LocalZone TraversalSubZone Bandwidth PerCall Limit: 1920**

Zones LocalZone TraversalSubZone Bandwidth PerCall Mode: <Limited/Unlimited/NoBandwidth>

Determines whether there is a limit on the bandwidth of any one traversal call being handled by the VCS.
NoBandwidth: no bandwidth available. No traversal calls can be made.
Default: Unlimited

Example: **xConfiguration Zones LocalZone TraversalSubZone Bandwidth PerCall Mode: Limited**

Command reference - xConfiguration

Zones LocalZone TraversalSubZone Bandwidth Total Limit: <1..100000000>

Specifies the total bandwidth (in kbps) allowed for all traversal calls being handled by the VCS (applies only if the mode is set to **Limited**).
Default: 500000

Example: **xConfiguration Zones LocalZone TraversalSubZone Bandwidth Total Limit: 500000**

Zones LocalZone TraversalSubZone Bandwidth Total Mode: <Limited/Unlimited/NoBandwidth>

Determines whether or not there is a limit to the total bandwidth of all traversal calls being handled by the VCS.

NoBandwidth: no bandwidth available. No traversal calls can be made.

Default: Unlimited

Example: **xConfiguration Zones LocalZone TraversalSubZone Bandwidth Total Mode: Limited**

Zones Policy SearchRules Rule [1..2000] Description: <S: 0..64>

A free-form description of the search rule.

Example: **xConfiguration Zones Policy SearchRules Rule 1 Description: "Send query to the DNS zone"**

Zones Policy SearchRules Rule [1..2000] Mode: <AliasPatternMatch/AnyAlias/AnyIPAddress>

Determines whether a query is sent to the target zone.

AliasPatternMatch: queries the zone only if the alias matches the corresponding pattern type and string.

AnyAlias: queries the zone for any alias (but not IP address).

AnyIPAddress: queries the zone for any given IP address (but not alias).

Default: AnyAlias

Example: **xConfiguration Zones Policy SearchRules Rule 1 Mode: AnyAlias**

Zones Policy SearchRules Rule [1..2000] Name: <S: 0..50>

Descriptive name for the search rule.

Example: **xConfiguration Zones Policy SearchRules Rule 1 Name: "DNS lookup"**

Zones Policy SearchRules Rule [1..2000] Pattern Behavior: <Strip/Leave/Replace>

Determines whether the matched part of the alias is modified before being sent to the target zone. (Applies to **Alias Pattern Match** mode only.)

Leave: the alias is not modified.

Strip: the matching prefix or suffix is removed from the alias.

Replace: the matching part of the alias is substituted with the text in the replace string.

Default: Strip

Example: **xConfiguration Zones Policy SearchRules Rule 1 Pattern Behavior: Strip**

Zones Policy SearchRules Rule [1..2000] Pattern Replace: <S: 0..60>

The string to substitute for the part of the alias that matches the pattern. (Applies to **Replace** pattern behavior only.)

Example: **xConfiguration Zones Policy SearchRules Rule 1 Pattern Replace: "@example.net"**

Zones Policy SearchRules Rule [1..2000] Pattern String: <S: 0..60>

The pattern against which the alias is compared. (Applies to **Alias Pattern Match** mode only.)

Example: **xConfiguration Zones Policy SearchRules Rule 1 Pattern String: "@example.com"**

Zones Policy SearchRules Rule [1..2000] Pattern Type: <Exact/Prefix/Suffix/Regex>

How the pattern string must match the alias for the rule to be applied. (Applies to **Alias Pattern Match** mode only.)

Exact: the entire string must exactly match the alias character for character.

Prefix: the string must appear at the beginning of the alias.

Suffix: the string must appear at the end of the alias.

Regex: the string is treated as a regular expression.

Default: Prefix

Example: **xConfiguration Zones Policy SearchRules Rule 1 Pattern Type: Suffix**

Zones Policy SearchRules Rule [1..2000] Priority: <1..65534>

The order in the search process that this rule is applied, when compared to the priority of the other search rules. All Priority 1 search rules are applied first, followed by all Priority 2 search rules, and so on.

Default: 100

Example: **xConfiguration Zones Policy SearchRules Rule 1 Priority: 100**

Zones Policy SearchRules Rule [1..2000] Progress: <Continue/Stop>

Specifies the ongoing search behavior if the alias matches this search rule. If 'stop' is selected, any rules with the same priority level as this rule are still applied.

Continue: continue applying the remaining search rules (in priority order) until the endpoint identified by the alias is found.

Stop: do not apply any more search rules, even if the endpoint identified by the alias is not found in the target zone.

Default: Continue

Example: **xConfiguration Zones Policy SearchRules Rule 1 Progress: Continue**

Zones Policy SearchRules Rule [1..2000] Source: <Any/AllZones/LocalZone>

The sources of the requests for which this rule applies.

Any: locally registered devices, neighbor or traversal zones, and any non-registered devices.

AllZones: locally registered devices plus neighbor or traversal zones.

LocalZone: locally registered devices only.

Default: Any

Example: **xConfiguration Zones Policy SearchRules Rule 1 Source: Any**

Zones Policy SearchRules Rule [1..2000] State: <Enabled/Disabled>

Indicates if the search rule is enabled or disabled. Disabled search rules are ignored.

Default: Enabled

Example: **xConfiguration Zones Policy SearchRules Rule 1 State: Enabled**

Command reference - xConfiguration

Zones Policy SearchRules Rule [1..2000] Target ZoneName: <S: 0..50>

The zone to query if the alias matches the search rule.

Example: **xConfiguration Zones Policy SearchRules Rule 1 Target ZoneName: "Sales Office"**

Zones Zone [1..1000] DNS IncludeAddressRecord: <On/Off>

Determines whether, if no NAPTR (SIP) or SRV (SIP and H.323) records have been found for the dialed alias via this zone, the VCS will then query for A and AAAA DNS Records.
Default: Off

Example: **xConfiguration Zones Zone 1 DNS IncludeAddressRecord: Off**

Zones Zone [1..1000] DNS Interworking SIP Audio DefaultCodec: <G711u/G711a/G722_48/G722_56/G722_64/G722_1_16/G722_1_24/G722_1_32/G722_1_48/G723_1/G728/G729/AALCD_48/AALCD_56/AALCD_64/AMR>

Specifies which audio codec to use when empty INVITES are not allowed.
Default: G711u

Example: **xConfiguration Zones Zone 1 DNS Interworking SIP Audio DefaultCodec: G711u**

Zones Zone [1..1000] DNS Interworking SIP EmptyInviteAllowed: <On/Off>

Determines whether the VCS will generate a SIP INVITE message with no SDP to send to this zone. INVITES with no SDP mean that the destination device is asked to initiate the codec selection, and are used when the call has been interworked locally from H.323.

On: SIP INVITES with no SDP will be generated and sent to this neighbor.

Off: SIP INVITES will be generated and a pre-configured SDP will be inserted before the INVITES are sent to this neighbor.

Default: On

Example: **xConfiguration Zones Zone 1 DNS Interworking SIP EmptyInviteAllowed: On**

Zones Zone [1..1000] DNS Interworking SIP Video DefaultBitrate: <64..2048>

Specifies which video bitrate to use when empty INVITES are not allowed.
Default: 384

Example: **xConfiguration Zones Zone 1 DNS Interworking SIP Video DefaultBitrate: 384**

Zones Zone [1..1000] DNS Interworking SIP Video DefaultCodec: <None/H261/H263/H263p/H263pp/H264>

Specifies which video codec to use when empty INVITES are not allowed.
Default: H263

Example: **xConfiguration Zones Zone 1 DNS Interworking SIP Video DefaultCodec: H263**

Zones Zone [1..1000] DNS Interworking SIP Video DefaultResolution: <None/QCIF/CIF/4CIF/SIF/4SIF/VGA/SVGA/XGA>

Specifies which video resolution to use when empty INVITES are not allowed.
Default: CIF

Example: **xConfiguration Zones Zone 1 DNS Interworking SIP Video DefaultResolution: CIF**

Zones Zone [1..1000] DNS SIP Duo Video Filter Mode: <On/Off>

Determines whether INVITE requests sent to this zone filter out Duo Video. This option may be required to enable interoperability with SIP devices that do not support Duo Video.

On: the second video line in any outgoing INVITE request is removed.

Off: INVITE requests are not modified.

Default: Off

Example: **xConfiguration Zones Zone 1 DNS SIP Duo Video Filter Mode: Off**

Zones Zone [1..1000] DNS SIP Poison Mode: <On/Off>

Determines whether SIP requests sent out to this zone will be "poisoned" such that if they are received by the local VCS again they will be rejected.

On: SIP requests sent out via this zone that are received again by this VCS will be rejected.

Off: SIP requests sent out via this zone that are received by this VCS again will be processed as normal.

Default: Off

Example: **xConfiguration Zones Zone 1 DNS SIP Poison Mode: Off**

Zones Zone [1..1000] DNS SIP Record Route Address Type: <IP/Hostname>

Controls whether the VCS uses its IP address or host name in the record-route or path headers of outgoing SIP requests to this zone.

Note: setting this value to **Hostname** also requires a valid DNS local host name to be configured on the VCS.

Default: IP

Example: **xConfiguration Zones Zone [1..1000] DNS SIP Record Route Address Type: IP**

Zones Zone [1..1000] DNS SIP SDP Attribute Line Limit Length: <80..65535>

If SIP SDP attribute line limit mode is set to On, sets the maximum line length of a=fmtp SDP lines.

Default: 130

Example: **xConfiguration Zones Zone 1 DNS SIP SDP Attribute Line Limit Length: 130**

Zones Zone [1..1000] DNS SIP SDP Attribute Line Limit Mode: <On/Off>

Determines whether requests containing SDP sent out to this zone will have the length of a=fmtp lines restricted.

On: the length will be truncated to the maximum length specified by the SIP SDP attribute line limit length setting.

Off: the length will not be truncated.

Example: **xConfiguration Zones Zone 1 DNS SIP SDP Attribute Line Limit Mode: Off**

Zones Zone [1..1000] DNS SIP SearchAutoResponse: <On/Off>

Determines what happens when the VCS receives a SIP search that originated as an H.323 search, destined for this zone.

Off: a SIP OPTION message will be sent to the zone.

On: searches will be responded to automatically, without being forwarded to the zone..

Default: Off

Example: **xConfiguration Zones Zone 1 DNS SIP SearchAutoResponse: Off**

Command reference - xConfiguration

Zones Zone [1..1000] DNS SIP TLS Verify Mode: <On/Off>

Controls X.509 certificate checking between this VCS and the destination system server returned by the DNS lookup. When enabled, the domain name submitted to the DNS lookup must be contained within the server's X.509 certificate (in either the Subject Common Name or the Subject Alternative Name attributes).

Default: Off

Example: **xConfiguration Zones Zone 1 DNS SIP TLS Verify Mode: On**

Zones Zone [1..1000] DNS SIP UDP BFCP Filter Mode: <On/Off>

Determines whether INVITE requests sent to this zone filter out UDP/BFCP. This option may be required to enable interoperability with SIP devices that do not support the UDP/BFCP protocol.

On: any media line referring to the UDP/BFCP protocol is replaced with TCP/BFCP and disabled.

Off: INVITE requests are not modified.

Default: Off

Example: **xConfiguration Zones Zone 1 DNS SIP UDP BFCP Filter Mode: Off**

Zones Zone [1..1000] DNS ZoneProfile: <Default/Custom/MicrosoftOCS2007/CiscoUnifiedCommunicationsManager/NortelCS1000/AdvancedMediaGateway>

Determines how the zone's advanced settings are configured.

Default: uses the factory defaults.

Custom: allows you to configure each setting individually.

Preconfigured profiles: alternatively, choose one of the preconfigured profiles to automatically use the appropriate settings required for connections to that type of system.

Default: Default

Example: **xConfiguration Zones Zone 1 DNS ZoneProfile: Default**

Zones Zone [1..1000] ENUM DNSSuffix: <S: 0, 128>

Specifies the DNS zone to be appended to the transformed E.164 number to create an ENUM host name which this zone is then queried for.

Example: **xConfiguration Zones Zone 2 ENUM DNSSuffix: "e164.arpa"**

Zones Zone [1..1000] H323 Mode: <On/Off>

Determines whether H.323 calls will be allowed to and from this zone.

Default: On

Example: **xConfiguration Zones Zone 2 H323 Mode: On**

Zones Zone [1..1000] HopCount: <1..255>

Specifies the hop count to be used when sending an alias search request to this zone. Note: if the search request was received from another zone and already has a hop count assigned, the lower of the two values will be used.

Default: 15

Example: **xConfiguration Zones Zone 2 HopCount: 15**

Zones Zone [1..1000] Name: <S: 1, 50>

Assigns a name to this zone.

Example: **xConfiguration Zones Zone 3 Name: "UK Sales Office"**

Command reference - xConfiguration

Zones Zone [1..1000] Neighbor AdvancedMediaGateway Mode: <On/Off>

Controls whether calls to or from this zone will use an Advanced Media Gateway.
Default: Off

Example: **xConfiguration Zones Zone 3 Neighbor AdvancedMediaGateway Mode: On**

Zones Zone [1..1000] Neighbor H323 Port: <1024..65534>

Specifies the port on the neighbor to be used for H.323 calls to and from this VCS.
Default: 1719

Example: **xConfiguration Zones Zone 3 Neighbor H323 Port: 1719**

Zones Zone [1..1000] Neighbor Interworking SIP Audio DefaultCodec: <G711u/G711a/G722_48/G722_56/G722_64/G722_1_16/G722_1_24/G722_1_32/G722_1_48/G723_1/G728/G729/AALCD_48/AALCD_56/AALCD_64/AMR>

Specifies which audio codec to use when empty INVITES are not allowed.
Default: G711u

Example: **xConfiguration Zones Zone 3 Neighbor Interworking SIP Audio DefaultCodec: G711u**

Zones Zone [1..1000] Neighbor Interworking SIP EmptyInviteAllowed: <On/Off>

Determines whether the VCS will generate a SIP INVITE message with no SDP to send to this zone. INVITES with no SDP mean that the destination device is asked to initiate the codec selection, and are used when the call has been interworked locally from H.323.

On: SIP INVITES with no SDP will be generated and sent to this neighbor.

Off: SIP INVITES will be generated and a pre-configured SDP will be inserted before the INVITES are sent to this neighbor..

Default: On

Example: **xConfiguration Zones Zone 3 Neighbor Interworking SIP EmptyInviteAllowed: On**

Zones Zone [1..1000] Neighbor Interworking SIP Search Strategy: <Options/Info>

Determines how the VCS will search for SIP endpoints when interworking an H.323 call.

Options: the VCS will send an OPTIONS request.

Info: the VCS will send an INFO request.

Default: Options

Example: **xConfiguration Zones Zone 3 Neighbor Interworking SIP Search Strategy: Options**

Zones Zone [1..1000] Neighbor Interworking SIP Video DefaultBitrate: <64..2048>

Specifies which video bitrate to use when empty INVITES are not allowed.
Default: 384

Example: **xConfiguration Zones Zone 3 Neighbor Interworking SIP Video DefaultBitrate: 384**

Zones Zone [1..1000] Neighbor Interworking SIP Video DefaultCodec: <None/H261/H263/H263p/H263pp/H264>

Specifies which video codec to use when empty INVITES are not allowed.
Default: H263

Example: **xConfiguration Zones Zone 3 Neighbor Interworking SIP Video DefaultCodec: H263**

Zones Zone [1..1000] Neighbor Interworking SIP Video DefaultResolution: <None/QCIF/CIF/4CIF/SIF/4SIF/VGA/SVGA/XGA>

Specifies which video resolution to use when empty INVITEs are not allowed.

Default: CIF

Example: **xConfiguration Zones Zone 3 Neighbor Interworking SIP Video DefaultResolution: CIF**

Zones Zone [1..1000] Neighbor Peer [1..6] Address: <S:0..128>

Specifies the IP address or Fully Qualified Domain Name (FQDN) of the neighbor. If the neighbor zone is a VCS cluster, this will be one of the peers in that cluster.

Example: **xConfiguration Zones Zone 3 Neighbor Peer 1 Address: "192.44.0.18"**

Zones Zone [1..1000] Neighbor Registrations: <Allow/Deny>

Controls whether proxied SIP registrations routed through this zone are accepted.

Default: Allow

Example: **xConfiguration Zones Zone 3 Neighbor Registrations: Allow**

Zones Zone [1..1000] Neighbor SIP Authentication Trust Mode: <On/Off>

Controls whether authenticated SIP messages (ones containing a P-Asserted-Identity header) from this zone are trusted.

On: messages are trusted without further challenge.

Off: messages are challenged for authentication.

Default: Off

Example: **xConfiguration Zones Zone 3 Neighbor SIP Authentication Trust Mode: On**

Zones Zone [1..1000] Neighbor SIP Duo Video Filter Mode: <On/Off>

Determines whether INVITE requests sent to this zone filter out Duo Video. This option may be required to enable interoperability with SIP devices that do not support Duo Video.

On: the second video line in any outgoing INVITE request is removed.

Off: INVITE requests are not modified.

Default: Off

Example: **xConfiguration Zones Zone 1 Neighbor SIP Duo Video Filter Mode: Off**

Zones Zone [1..1000] Neighbor SIP Encryption Mode: <Auto/Off>

Determines whether or not the VCS will allow encrypted SIP calls on this zone.

Auto: SIP calls will be encrypted if a secure SIP transport (TLS) is used.

Off: SIP calls will never be encrypted.

Default: Auto

Example: **xConfiguration Zones Zone 3 Neighbor SIP Encryption Mode: Auto**

Zones Zone [1..1000] Neighbor SIP MIME Strip Mode: <On/Off>

Controls whether multipart MIME stripping is performed on requests from this zone. This must be set to On for connections to a Microsoft Office Communications Server 2007.

Default: Off

Example: **xConfiguration Zones Zone 3 Neighbor SIP MIME Strip Mode: Off**

Zones Zone [1..1000] Neighbor SIP MediaRouting Mode: <Auto/Signaled/Latching>

Specifies how the VCS handles the media for calls to and from this neighbor, and where it will forward the media destined for this neighbor.

Signaled: the media is always taken for calls to and from this neighbor. It will be forwarded as signaled in the SDP received from this neighbor.

Latching: the media is always taken for calls to and from this neighbor. It will be forwarded to the IP address and port from which media from this neighbor is received.

Auto: media is only taken if the call is a traversal call. If this neighbor is behind a NAT the VCS will forward the media to the IP address and port from which media from this zone is received (latching). Otherwise it will forward the media to the IP address and port signaled in the SDP (signaled).

Default: Auto.

Example: **xConfiguration Zones Zone 3 Neighbor SIP MediaRouting Mode: Auto**

Zones Zone [1..1000] Neighbor SIP Poison Mode: <On/Off>

Determines whether SIP requests sent out to this zone will be "poisoned" such that if they are received by the local VCS again they will be rejected.

On: SIP requests sent out via this zone that are received again by this VCS will be rejected.

Off: SIP requests sent out via this zone that are received by this VCS again will be processed as normal.

Default: Off

Example: **xConfiguration Zones Zone 3 Neighbor SIP Poison Mode: Off**

Zones Zone [1..1000] Neighbor SIP Port: <1024..65534>

Specifies the port on the neighbor to be used for SIP calls to and from this VCS.

Default: 5061

Example: **xConfiguration Zones Zone 3 Neighbor SIP Port: 5061**

Zones Zone [1..1000] Neighbor SIP ProxyRequire Strip List: <S: 0..255>

A comma separated list of option tags to search for and remove from Proxy-Require headers in SIP requests received from this zone. By default, no option tags are specified.

Example: **xConfiguration Zones Zone 3 Neighbor SIP ProxyRequire Strip List: "com.example.something,com.example.somethingelse"**

Zones Zone [1..1000] Neighbor SIP Record Route Address Type: <IP/Hostname>

Controls whether the VCS uses its IP address or host name in the record-route or path headers of outgoing SIP requests to this zone.

Note: setting this value to **Hostname** also requires a valid DNS local host name to be configured on the VCS.

Default: IP

Example: **xConfiguration Zones Zone [1..1000] Neighbor SIP Record Route Address Type: IP**

Zones Zone [1..1000] Neighbor SIP SDP Attribute Line Limit Length: <80..65535>

If SIP SDP attribute line limit mode is set to On, sets the maximum line length of a=fmtp SDP lines.

Default: 130

Example: **xConfiguration Zones Zone 3 Neighbor SIP SDP Attribute Line Limit Length: 130**

Zones Zone [1..1000] Neighbor SIP SDP Attribute Line Limit Mode: <On/Off>

Determines whether requests containing SDP sent out to this zone will have the length of a=fmtp lines restricted.

On: the length will be truncated to the maximum length specified by the SIP SDP attribute line limit length setting.

Off: the length will not be truncated.

Default: Off

Example: **xConfiguration Zones Zone 3 Neighbor SIP SDP Attribute Line Limit Mode: Off**

Zones Zone [1..1000] Neighbor SIP SearchAutoResponse: <On/Off>

Determines what happens when the VCS receives a SIP search that originated as an H.323 search, destined for this zone.

Off: a SIP OPTION message will be sent to the zone.

On: searches will be responded to automatically, without being forwarded to the zone..

Default: Off

Example: **xConfiguration Zones Zone 3 Neighbor SIP SearchAutoResponse: Off**

Zones Zone [1..1000] Neighbor SIP TLS Verify Mode: <On/Off>

Controls X.509 certificate checking and mutual authentication for inbound and outbound connections between this VCS and the neighbor system. When enabled, the neighbor system's FQDN or IP address, as specified in the Peer address field, must be contained within the neighbor's X.509 certificate (in either the Subject Common Name or the Subject Alternative Name attributes).

Default: Off

Example: **xConfiguration Zones Zone 3 Neighbor SIP TLS Verify Mode: On**

Zones Zone [1..1000] Neighbor SIP Transport: <UDP/TCP/TLS>

Determines which transport type will be used for SIP calls to and from this neighbor.

Default: TLS

Example: **xConfiguration Zones Zone 3 Neighbor SIP Transport: TLS**

Zones Zone [1..1000] Neighbor SIP UDP BFCP Filter Mode: <On/Off>

Determines whether INVITE requests sent to this zone filter out UDP/BFCP. This option may be required to enable interoperability with SIP devices that do not support the UDP/BFCP protocol.

On: any media line referring to the UDP/BFCP protocol is replaced with TCP/BFCP and disabled.

Off: INVITE requests are not modified.

Default: Off

Example: **xConfiguration Zones Zone 1 Neighbor SIP UDP BFCP Filter Mode: Off**

Zones Zone [1..1000] Neighbor SIP UPDATE Strip Mode: <On/Off>

Determines whether or not the VCS will strip the UPDATE method from the Allow header of all requests and responses received from this zone.

Default: Off

Example: **xConfiguration Zones Zone 3 Neighbor SIP UPDATE Strip Mode: Off**

Zones Zone [1..1000] Neighbor ZoneProfile: <Default/Custom/MicrosoftOCS2007/CiscoUnifiedCommunicationsManager/NortelCS1000/AdvancedMediaGateway>

Determines how the zone's advanced settings are configured.

Default: uses the factory defaults.

Custom: allows you to configure each setting individually.

Preconfigured profiles: alternatively, choose one of the preconfigured profiles to automatically use the appropriate settings required for connections to that type of system.

Default: Default

Example: **Zones Zone 3 Neighbor ZoneProfile: Default**

Zones Zone [1..1000] SIP Mode: <On/Off>

Determines whether SIP calls will be allowed to and from this zone.

Default: On

Example: **xConfiguration Zones Zone 3 SIP Mode: On**

Zones Zone [1..1000] TraversalClient Authentication Password: <S: 0..215>

The password used by the VCS when connecting to the traversal server. The maximum plaintext length is 128 characters, which is then encrypted.

Example: **xConfiguration Zones Zone 1 TraversalClient Authentication Password: "password123"**

Zones Zone [1..1000] TraversalClient Authentication UserName: <S: 0..128>

The user name used by the VCS when connecting to the traversal server.

Example: **xConfiguration Zones Zone 1 TraversalClient Authentication UserName: "clientname"**

Zones Zone [1..1000] TraversalClient H323 Port: <1024..65534>

Specifies the port on the traversal server to be used for H.323 firewall traversal calls from this VCS. If the traversal server is a VCS Expressway, this must be the port number that has been configured on the VCS Expressway's traversal server zone associated with this VCS.

Example: **xConfiguration Zones Zone 4 TraversalClient H323 Port: 2777**

Zones Zone [1..1000] TraversalClient H323 Protocol: <Assent/H46018>

Determines which of the two firewall traversal protocols will be used for calls to and from the traversal server.

Note: the same protocol must be set on the server for calls to and from this traversal client.

Default: Assent

Example: **xConfiguration Zones Zone 4 TraversalClient H323 Protocol: Assent**

Zones Zone [1..1000] TraversalClient Peer [1..6] Address: <S:0..128>

Specifies the IP address or Fully Qualified Domain Name (FQDN) of the traversal server. If the traversal server is a VCS Expressway cluster, this will be one of the peers in that cluster.

Example: **xConfiguration Zones Zone 4 TraversalClient Peer 1 Address: "10.192.168.1"**

Zones Zone [1..1000] TraversalClient Registrations: <Allow/Deny>

Controls whether proxied SIP registrations routed through this zone are accepted.

Default: Allow

Example: **xConfiguration Zones Zone 4 TraversalClient Registrations: Allow**

Zones Zone [1..1000] TraversalClient RetryInterval: <1..65534>

Specifies the interval (in seconds) with which a failed attempt to establish a connection to the traversal server should be retried.
Default: 120

Example: **xConfiguration Zones Zone 4 TraversalClient RetryInterval: 120**

Zones Zone [1..1000] TraversalClient SIP Poison Mode: <On/Off>

Determines whether SIP requests sent out to this zone will be "poisoned" such that if they are received by the local VCS again they will be rejected.

On: SIP requests sent out via this zone that are received again by this VCS will be rejected.

Off: SIP requests sent out via this zone that are received by this VCS again will be processed as normal.

Default: Off

Example: **xConfiguration Zones Zone 4 TraversalClient SIP Poison Mode: Off**

Zones Zone [1..1000] TraversalClient SIP Port: <1024..65534>

Specifies the port on the traversal server to be used for SIP calls from this VCS. If your traversal server is a VCS Expressway, this must be the port number that has been configured in the traversal server zone for this VCS.

Example: **xConfiguration Zones Zone 4 TraversalClient SIP Port: 5061**

Zones Zone [1..1000] TraversalClient SIP TLS Verify Mode: <On/Off>

Controls X.509 certificate checking and mutual authentication between this VCS and the traversal server. When enabled, the server's FQDN or IP address, as specified in the Peer address field, must be contained within the server's X.509 certificate (in either the Subject Common Name or the Subject Alternative Name attributes).

Default: Off

Example: **xConfiguration Zones Zone 4 TraversalClient SIP TLS Verify Mode: On**

Zones Zone [1..1000] TraversalClient SIP Transport: <TCP/TLS>

Determines which transport type will be used for SIP calls to and from the traversal server.

Default: TLS

Example: **xConfiguration Zones Zone 4 TraversalClient SIP Transport: TLS**

Zones Zone [1..1000] TraversalServer Authentication UserName: <S: 1, 128>

The name used by the traversal client when authenticating with the traversal server. If the traversal client is a VCS, this must be the VCS's authentication user name. If the traversal client is a gatekeeper, this must be the gatekeeper's System Name. For other types of traversal clients, refer to the VCS Admin Guide for further information.

Example: **xConfiguration Zones Zone 5 TraversalServer Authentication UserName: "User123"**

Zones Zone [1..1000] TraversalServer H323 H46019 Demultiplexing Mode: <On/Off>

Determines whether the VCS will operate in demultiplexing mode for calls from the traversal client.

On: allows use of the same two ports for all calls.

Off: each call will use a separate pair of ports for media.

Default: Off

Example: **xConfiguration Zones Zone 5 TraversalServer H323 H46019 Demultiplexing Mode: Off**

Command reference - xConfiguration

Zones Zone [1..1000] TraversalServer H323 Port: <1024..65534>

Specifies the port on the VCS being used for H.323 firewall traversal from this traversal client.
Default: 6001, incrementing by 1 for each new zone.

Example: **xConfiguration Zones Zone 5 TraversalServer H323 Port: 2777**

Zones Zone [1..1000] TraversalServer H323 Protocol: <Assent/H46018>

Determines which of the two firewall traversal protocols will be used for calls to and from the traversal client.
Note: the same protocol must be set on the client for calls to and from this traversal server.
Default: Assent

Example: **xConfiguration Zones Zone 5 TraversalServer H323 Protocol: Assent**

Zones Zone [1..1000] TraversalServer Registrations: <Allow/Deny>

Controls whether proxied SIP registrations routed through this zone are accepted.
Default: Allow

Example: **xConfiguration Zones Zone 5 TraversalServer Registrations: Allow**

Zones Zone [1..1000] TraversalServer SIP Poison Mode: <On/Off>

Determines whether SIP requests sent out to this zone will be "poisoned" such that if they are received by the local VCS again they will be rejected.

On: SIP requests sent out via this zone that are received again by this VCS will be rejected.

Off: SIP requests sent out via this zone that are received by this VCS again will be processed as normal.

Default: Off

Example: **xConfiguration Zones Zone 5 TraversalServer SIP Poison Mode: Off**

Zones Zone [1..1000] TraversalServer SIP Port: <1024..65534>

Specifies the port on the VCS being used for SIP firewall traversal from this traversal client.
Default: 7001, incrementing by 1 for each new zone.

Example: **xConfiguration Zones Zone 5 TraversalServer SIP Port: 5061**

Zones Zone [1..1000] TraversalServer SIP TLS Verify Mode: <On/Off>

Controls X.509 certificate checking and mutual authentication between this VCS and the traversal client. If enabled, a TLS verify subject name must be specified.
Default: Off

Example: **xConfiguration Zones Zone 5 TraversalServer SIP TLS Verify Mode: On**

Zones Zone [1..1000] TraversalServer SIP TLS Verify Subject Name: <S: 0..128>

The certificate holder's name to look for in the traversal client's X.509 certificate (must be in either the Subject Common Name or the Subject Alternative Name attributes).

Example: **xConfiguration Zones Zone 5 TraversalServer SIP TLS Verify Subject Name: "myclientname"**

Command reference - xConfiguration

Zones Zone [1..1000] TraversalServer SIP Transport: <TCP/TLS>

Determines which of the two transport types will be used for SIP calls between the traversal client and VCS.

Default: TLS

Example: **xConfiguration Zones Zone 5 TraversalServer SIP Transport: TLS**

Zones Zone [1..1000] TraversalServer TCPProbe KeepAliveInterval: <1..65534>

Sets the interval (in seconds) with which the traversal client will send a TCP probe to the VCS once a call is established, in order to keep the firewall's NAT bindings open.

Default: 20

Example: **xConfiguration Zones Zone 5 TraversalServer TCPProbe KeepAliveInterval: 20**

Zones Zone [1..1000] TraversalServer TCPProbe RetryCount: <1..65534>

Sets the number of times the traversal client will attempt to send a TCP probe to the VCS.

Default: 5

Example: **xConfiguration Zones Zone 5 TraversalServer TCPProbe RetryCount: 5**

Zones Zone [1..1000] TraversalServer TCPProbe RetryInterval: <1..65534>

Sets the frequency (in seconds) with which the traversal client will send a TCP probe to the VCS.

Default: 2

Example: **xConfiguration Zones Zone 5 TraversalServer TCPProbe RetryInterval: 2**

Zones Zone [1..1000] TraversalServer UDPProbe KeepAliveInterval: <1..65534>

Sets the interval (in seconds) with which the traversal client will send a UDP probe to the VCS once a call is established, in order to keep the firewall's NAT bindings open.

Default: 20

Example: **xConfiguration Zones Zone 5 TraversalServer UDPProbe KeepAliveInterval: 20**

Zones Zone [1..1000] TraversalServer UDPProbe RetryCount: <1..65534>

Sets the number of times the traversal client will attempt to send a UDP probe to the VCS.

Default: 5

Example: **xConfiguration Zones Zone 5 TraversalServer UDPProbe RetryCount: 5**

Zones Zone [1..1000] TraversalServer UDPProbe RetryInterval: <1..65534>

Sets the frequency (in seconds) with which the traversal client will send a UDP probe to the VCS.

Default: 2

Example: **xConfiguration Zones Zone 5 TraversalServer UDPProbe RetryInterval: 2**

Zones Zone [1..1000] Type: <Neighbor/TraversalClient/TraversalServer/ENUM/DNS>

Determines the nature of the specified zone, in relation to the local VCS.

Neighbor: the new zone will be a neighbor of the local VCS.

TraversalClient: there is a firewall between the zones, and the local VCS is a traversal client of the new zone.

TraversalServer: there is a firewall between the zones and the local VCS is a traversal server for the new zone.

ENUM: the new zone contains endpoints discoverable by ENUM lookup.

DNS: the new zone contains endpoints discoverable by DNS lookup.

Example: `xConfiguration Zones Zone 3 Type: Neighbor`

Overview

The **xCommand** group of commands are used to add and delete items and issue system commands.

The following pages list all the **xCommand** commands currently available on the VCS.

To issue a command, type the command as shown, followed by one or more of the given parameters and values. The valid values for each parameter are indicated in the angle brackets following each parameter; these are explained opposite.

To obtain information about using each of the **xCommand** commands from within the CLI:

- type **xCommand** or **xCommand ?** to return all current **xCommand** commands available on the VCS.
- type **xCommand ??** to return all current **xCommand** commands available on the VCS, along with a description of each command, a list of its parameters, and for each parameter its valuespaces and description.
- type **xCommand <command> ?** to return a description of the command, a list of its parameters, and for each parameter its valuespaces and description.

The valid value for this parameter is an integer. The minimum and maximum values are shown within the angle brackets.

The valid value for this parameter is a string. The minimum and maximum number of characters is shown after the **S**.

When issuing this command, the string must be typed in double quotes.

The valid values for this parameter are one of the options shown within the angle brackets.

(r) indicates that this is a required parameter. The (r) is not part of the command.

AdminAccountDelete

Deletes an administrator account.

AdminAccountId(r): <1..15>

The index of the administrator account to be deleted.

Example: **xCommand AdminAccountDelete AdminAccountId: 1**

AdminLoginGroupAdd

Creates a new administrator login group.

Name(r): <S: 0..128>

Defines the name of an administrator group that determines which access rights memb

Access(r): <None/ReadOnly/ReadWrite/Auditor>

Defines the access level for members of the specified administrator group. **None**: no access viewed and changed. **Auditor**: allows access to the Event Log, Configuration Log and the Default: ReadWrite

Example: **xCommand AdminLoginGroupAdd Name: "VCS" Access: ReadWrite**

AdminLoginGroupDelete

Deletes an administrator login group.

AdminLoginGroupId(r): <1..30>

The index of the administrator login group to be deleted.

Example: **xCommand AdminLoginGroupDelete AdminLoginGroupId: 1**

AMGWPolicyRuleAdd

Adds and configures a new Advanced Media Gateway policy rule.

Name(r): <S: 1..50>

Assigns a name to this Advanced Media Gateway policy rule.

Description: <S: 0..64>

A free-form description of the membership rule.

Example: **xCommand AMGWPolicyRuleAdd Name: "Deny branch calls" Description: "Deny all calls to branch office"**

AMGWPolicyRuleDelete

Deletes an Advanced Media Gateway policy rule.

AMGWPolicyRuleId(r): <1..200>

The index of the Advanced Media Gateway policy rule to be deleted.

Example: **xCommand AMGWPolicyRuleDelete AMGWPolicyRuleId: 1**

AdminAccountAdd

Creates a new administrator account.

Name(r): <S:0..25>

Defines the name of an administrator user who can login to the VCS web interface.

Password(r): <S:0..65>

Defines the password of an administrator user who can login to the VCS web interface. The maximum plaintext length is 16 characters, which will then be encrypted.

Access(r): <AccountDisabled/ReadOnly/ReadWrite/Auditor >

Defines the access level of an administrator user who can login to the VCS web interface. **AccountDisabled**: no access allowed. **ReadOnly**: configuration can only be viewed.

ReadWrite: configuration can be viewed and changed. **Auditor**: allows access to the Event Log, Configuration Log and the Overview page only.

Default: ReadWrite

Example: **xCommand AdminAccountAdd Name: "guest" Password: "password123" Access: readonly**

AdminAccountDelete

Deletes an administrator account.

AdminAccountId(r): <1..15>

The index of the administrator account to be deleted.

Example: **xCommand AdminAccountDelete AdminAccountId: 1**

Command reference - xCommand

AdminLoginGroupAdd

Creates a new administrator login group.

Name(r): <S: 0..128>

Defines the name of an administrator group that determines which access rights members of the group have after they have been successfully authenticated to use the VCS.

Access(r): <None/ReadOnly/ReadWrite/Auditor>

Defines the access level for members of the specified administrator group. **None**: no access allowed. **ReadOnly**: configuration can only be viewed. **ReadWrite**: configuration can be viewed and changed. **Auditor**: allows access to the Event Log, Configuration Log and the Overview page only.

Default: ReadWrite

Example: **xCommand AdminLoginGroupAdd Name: "VCS" Access: ReadWrite**

AdminLoginGroupDelete

Deletes an administrator login group.

AdminLoginGroupId(r): <1..30>

The index of the administrator login group to be deleted.

Example: **xCommand AdminLoginGroupDelete AdminLoginGroupId: 1**

AllowListAdd

Adds an entry to the Allow List.

PatternString(r): <S: 1, 60>

Specifies an entry to be added to the Allow List. If one of an endpoint's aliases matches one of the patterns in the Allow List, the registration will be permitted.

PatternType: <Exact/Prefix/Suffix/Regex>

Specifies whether the entry in the Allow List is a prefix, suffix, regular expression, or must be matched exactly. **Exact**: the string must match the alias character for character. **Prefix**: the string must appear at the beginning of the alias. **Suffix**: the string must appear at the end of the alias. **Regex**: the string will be treated as a regular expression.

Default: Exact.

Description: <S: 0..64>

A free-form description of the Allow List rule.

Example: **xCommand AllowListAdd PatternString: "John.Smith@example.com" PatternType: Exact Description: "Allow John Smith"**

AllowListDelete

Deletes an entry from the Allow List.

AllowListId(r): <1..2500>

The index of the entry to be deleted.

Example: **xCommand AllowListDelete AllowListId: 2**

Boot

Reboots the VCS.

This command has no parameters.

Example: **xCommand boot**

CheckBandwidth

A diagnostic tool that returns the status and route (as a list of nodes and links) that a call of the specified type and bandwidth would take between two nodes.

Note that this command does not change any existing system configuration.

Node1(r): <S: 1, 50>

The subzone or zone from which the call originates.

Node2(r): <S: 1, 50>

The subzone or zone at which the call terminates.

Bandwidth(r): <1..100000000>

The requested bandwidth of the call (in kbps).

CallType(r): <Traversal/NonTraversal>

Whether the call type is Traversal or Non-traversal.

Example: **xCommand CheckBandwidth Node1: "DefaultSubzone" Node2: "UK Sales Office" Bandwidth: 512 CallType: nontraversal**

CheckPattern

A diagnostic tool that allows you to check the result of an alias transform (local or zone) before you configure it on the system.

Note that this command does not change any existing system configuration.

Target(r): <S: 1, 60>

The alias you want to use to test the pattern match or transform.

Pattern(r): <S: 1, 60>

The pattern against which the alias is compared.

Type(r): <Exact/Prefix/Suffix/Regex>

How the pattern string must match the alias for the pattern behavior to be applied.

Behavior(r): <Strip/Leave/Replace/AddPrefix/AddSuffix>

How the alias is modified.

Replace: <S: 0, 60>

The text string to use in conjunction with the selected Pattern behavior.

Example: **xCommand CheckPattern Target: "john.smith@example.net" Pattern: "@example.net" Type: "suffix" Behavior: replace Replace: "@example.com"**

CredentialAdd

Adds an entry to the local authentication database.

CredentialName(r): <S: 1, 128>

Defines the name for this entry in the local authentication database.

CredentialPassword(r): <S: 1, 128>

Defines the password for this entry in the local authentication database.

Example: **xCommand CredentialAdd CredentialName: "John Smith" CredentialPassword: "password123"**

CredentialDelete

Deletes an entry from the local authentication database.

CredentialId(r): <1..2500>

The index of the credential to be deleted.

Example: **xCommand CredentialDelete CredentialId: 2**

DefaultLinksAdd

Restores links between the Default Subzone, Traversal Subzone and the Default Zone.

This command has no parameters.

Example: **xCommand DefaultLinksAdd**

DefaultValuesSet

Resets system parameters to default values.

Level(r): <1..3>

The level of system parameters to be reset.

Level 1: resets most configuration items to their default value, with the exception of the Level 2 and Level 3 items.

Level 2: resets configuration items related to remote authentication, plus Level 1 items to their default value.

Level 3: resets all critical configuration items, plus Level 1 and Level 2 items to their default value. See the [Restoring default configuration](#) section for full details.

Example: **xCommand DefaultValuesSet Level: 1**

DenyListAdd

Adds an entry to the Deny List.

PatternString(r): <S: 1, 60>

Specifies an entry to be added to the Deny List. If one of an endpoint's aliases matches one of the patterns in the Deny List, the registration will not be permitted.

PatternType: <Exact/Prefix/Suffix/Regex>

Specifies whether the entry in the Deny List is a prefix, suffix, regular expression, or must be matched exactly. Exact: the string must match the alias character for character. Prefix: the string must appear at the beginning of the alias. Suffix: the string must appear at the end of the alias. Regex: the string will be treated as a regular expression.

Default: Exact.

Description: <S: 0..64>

A free-form description of the Deny List rule.

Example: **xCommand DenyListAdd PatternString: "sally.jones@example.com" PatternType: exact Description: "Deny Sally Jones"**

DenyListDelete

Deletes an entry from the Deny List.

DenyListId(r): <1..2500>

The index of the entry to be deleted.

Example: **xCommand DenyListDelete DenyListId: 2**

Command reference - xCommand

DisconnectCall

Disconnects a call.

Call: <1..900>

The index of the call to be disconnected.

CallSerialNumber: <S: 1, 255>

The serial number of the call to be disconnected.

Note: you must specify either a call index or call serial number when using this command.

Example: **xCommand DisconnectCall CallSerialNumber: "6d843434-211c-11b2-b35d-0010f30f521c"**

DomainAdd

Adds a SIP domain for which this VCS is authoritative.

DomainName(r): <S: 1, 128>

Specifies a domain for which this VCS is authoritative. The VCS will act as a SIP registrar and Presence Server for this domain, and will accept registration requests for any SIP endpoints attempting to register with an alias that includes this domain. The domain name can comprise multiple levels. Each level's name can only contain letters, digits and hyphens, with each level separated by a period (dot). A level name cannot start or end with a hyphen, and the final level name must start with a letter. An example valid domain name is "100.example-name.com".

Example: **xCommand DomainAdd DomainName: "example.com"**

DomainDelete

Deletes a domain.

DomainId(r): <1..20>

The index of the domain to be deleted.

Example: **xCommand DomainDelete DomainId: 2**

ExtAppStatusAdd

Allows another application running on the VCS to attach xstatus to the VCS XML xstatus tree.

Note: this command is intended for developer use only.

Name(r): <S:1..64>

Descriptive name for the external application whose status is being referenced.

Filename(r): <S:0..255>

XML file containing status that is to be attached for an external application.

Example: **xCommand ExtAppStatusAdd Name: "foo" Filename: "foo.xml"**

ExtAppStatusDelete

Deletes an external application status entry.
Note: this command is intended for developer use only.

Name(r): <S:1..64>
Descriptive name for the external application whose status is being referenced.

Example: **xCommand ExtAppStatusDelete Name: foo**

FeedbackDeregister

Deactivates a particular feedback request.

ID: <1..3>
The index of the feedback request to be deactivated.

Example: **xCommand FeedbackDeregister ID: 1**

FeedbackRegister

Activates notifications on the event or status change(s) described by the expression(s). Notifications are sent in XML format to the specified URL. Up to 15 expressions may be registered for each of 3 feedback IDs.

ID: <1..3>
The ID of this particular feedback request.

URL(r): <S: 1, 256>
The URL to which notifications are to be sent.

Expression.1..15: <S: 1, 256>
The events or status change to be notified. Valid Expressions are:

Status/Ethernet	Status/Calls	Event/CallDisconnected	Event/
Status/NTP	Status/Registrations	Event/CallFailure	Event/Bandwidth
Status/LDAP	Status/Zones	Event/RegistrationAdded	Event/Locate
Status/Feedback	Event/CallAttempt	Event/RegistrationRemoved	Event/ResourceUsage
Status/ExternalManager	Event/CallConnected	Event/RegistrationFailure	Event/AuthenticationFailure

Example: **xCommand FeedbackRegister ID: 1 URL: "http://192.168.0.1/submitfeedback/" Expression.1: "Status/Calls" Expression.2: "Event/CallAttempt"**

FindRegistration

Returns information about the registration associated with the specified alias. The alias must be registered on the VCS on which the command is issued.

Alias(r): <S: 1, 60>
The alias that you wish to find out about.

Example: **xCommand FindRegistration Alias: "john.smith@example.com"**

ForceConfigUpdate

Performs an **xCommand DefaultValueSet Level: 2** on the specified peer, and then forces the relevant configuration on the peer to be updated to match that of the cluster master.

PeerId: <1..6>

The index of the cluster peer to be updated.

Example: **xCommand ForceConfigUpdate PeerId: 1**

LinkAdd

Adds and configures a new link.

LinkName(r): <S: 1, 50>

Assigns a name to this link.

Node1: <S: 1, 50>

Specifies the first zone or subzone to which this link will be applied.

Node2: <S: 1, 50>

Specifies the second zone or subzone to which this link will be applied.

Pipe1: <S: 1, 50>

Specifies the first pipe to be associated with this link.

Pipe2: <S: 1, 50>

Specifies the second pipe to be associated with this link.

Example: **xCommand LinkAdd LinkName: "Subzone1 to UK" Node1: "Subzone1" Node2: "UK Sales Office" Pipe1: "512Kb ASDL"**

LinkDelete

Deletes a link.

LinkId(r): <1..3000>

The index of the link to be deleted.

Example: **xCommand LinkDelete LinkId: 2**

ListPresentities

Returns a list of all the presentities being watched by a particular subscriber.

Subscriber(r): <S:1..255>

The URI of the subscriber who is watching.

Example: **xCommand ListPresentities Subscriber: "john.smith@example.com"**

ListSubscribers

Returns a list of all subscribers who are watching for the presence information of a particular presentity.

Presentity(r): <S:1..255>

The URI of the presentity being watched.

Example: **xCommand ListSubscribers Presentity: "mary.jones@example.com"**

Locate

Runs the VCS's location algorithm to locate the endpoint identified by the given alias, searching locally, on neighbors, and on systems discovered through the DNS system, within the specified number of 'hops'. Results are reported back through the xFeedback mechanism, which must therefore be activated before issuing this command (e.g. **xFeedback register event/locate**).

Alias(r): <S: 1, 60>

The alias associated with the endpoint you wish to locate.

HopCount(r): <0..255>

The hop count to be used in the search.

Protocol(r): <H323/SIP>

The protocol used to initiate the search.

SourceZone: <S: 1..50>

The zone from which to simulate the search request. Choose from the Default Zone (an unknown remote system), the Local Zone (a locally registered endpoint) or any other configured neighbor, traversal client or traversal server zone.

Example: **xCommand Locate Alias: "john.smith@example.com" HopCount: 15 Protocol: SIP SourceZone: LocalZone**

Log

Sets logging levels on specific modules.

Note: this command is intended for developer use only.

Module: <S:0..255>

The name of the module.

TraceLevel: <Default/Error/Warn/Info/Debug/Trace>

The level of tracing to use on the specified module. **Default** returns the trace level to its default value.

Example: **xCommand Log Module: "foo" TraceLevel: Error**

LogPersist

Saves the current log levels so that they will persist over a restart.

This command has no parameters.

Example: **xCommand LogPersist**

OptionKeyAdd

Adds a new option key to the VCS. These are added to the VCS in order to add extra functionality, such as increasing the VCS's capacity. Contact your Cisco representative for further information.

Key(r): <S: 0, 90>

Specifies the option key of your software option.

Example: **xCommand OptionKeyAdd Key: "1X4757T5-1-60BAD5CD"**

Command reference - xCommand

OptionKeyDelete

Deletes a software option key from the VCS.

OptionKeyId(r): <1..64>

Specifies the ID of the software option to be deleted.

Example: **xCommand OptionKeyDelete OptionKeyId: 2**

PipeAdd

Adds and configures a new pipe.

PipeName(r): <S: 1, 50>

Assigns a name to this pipe.

TotalMode: <Unlimited/Limited/NoBandwidth>

Determines whether or not this pipe is enforcing total bandwidth restrictions. NoBandwidth: no bandwidth available; no calls can be made using this pipe. Default: Unlimited.

Total: <1..100000000>

If this pipe has limited bandwidth, sets the maximum bandwidth (in kbps) available at any one time on the pipe. Default: 500000.

PerCallMode: <Unlimited/Limited/NoBandwidth>

Determines whether or not this pipe is limiting the bandwidth of individual calls. NoBandwidth: no bandwidth available; no calls can be made using this pipe. Default: Unlimited.

PerCall: <1..100000000>

If this pipe has limited per-call bandwidth, sets the maximum amount of bandwidth (in kbps) available for any one call. Default: 1920.

Example: **xCommand PipeAdd PipeName: "512k ADSL" TotalMode: Limited Total: 512 PerCallMode: Limited PerCall: 128**

PipeDelete

Deletes a pipe.

PipeId(r): <1..1000>

The index of the pipe to be deleted.

Example: **xCommand PipeDelete PipeId: 2**

RemoveRegistration

Removes a registration from the VCS.

Registration: <1..3750>

The index of the registration to be removed.

RegistrationSerialNumber: <S: 1, 255>

The serial number of the registration to be removed.

Example: **xCommand RemoveRegistration RegistrationSerialNumber: "a761c4bc-25c9-11b2-a37f-0010f30f521c"**

Restart

Restarts the VCS without a full system reboot.

This command has no parameters.

Example: **xCommand Restart**

RouteAdd

Adds and configures a new IP route (also known as a static route).

Address(r): <S: 1, 39>

Specifies an IP address used in conjunction with the prefix length to determine the network to which this route applies.

Default: 32

PrefixLength(r): <1..128>

Specifies the number of bits of the IP address which must match when determining the network to which this route applies.

Gateway(r): <S: 1, 39>

Specifies the IP address of the gateway for this route.

Interface: <Auto/LAN1/LAN2>

Specifies the LAN interface to use for this route. Auto: the VCS will select the most appropriate interface to use.

Default: Auto

Example: **xCommand RouteAdd Address: "10.13.8.0" PrefixLength: 32 Gateway: "192.44.0.1"**

RouteDelete

Deletes a route.

RouteId(r): <1..50>

The index of the route to be deleted.

Example: **xCommand RouteDelete RouteId: 1**

SearchRuleAdd

Adds a new search rule to route searches and calls toward a zone.

Name(r): <S: 0..50>

Descriptive name for the search rule.

ZoneName: <S: 0..50>

The zone to query if the alias matches the search rule.

Description: <S: 0..64>

A free-form description of the search rule.

Example: **xCommand SearchRuleAdd Name: "DNS lookup" ZoneName: "Sales Office" Description: "Send query to the DNS zone"**

SearchRuleDelete

Deletes a search rule.

SearchRuleId(r): <1..2000>

The index of the search rule to be deleted.

Example: **xCommand SearchRuleDelete SearchRuleId: 1**

SecureModeOff

Turns secure mode off - removes all audit information that contains sensitive information, such as log files and call, status and login history records.

This command has no parameters.

Example: **xCommand SecureModeOff**

SecureModeOn

Turns secure mode on - certain features and login accounts will be unavailable.

This command has no parameters.

Example: **xCommand SecureModeOn**

SIPRouteAdd

Adds a route that will cause SIP messages matching the given criteria to be forwarded to the specified IP address and port.

Note: this command is intended for developer use only.

Method(r): <S:0..64>

SIP method to match to select this route (e.g. INVITE, SUBSCRIBE).

RequestLinePattern(r): <S:0..128>

Regular expression to match against the SIP request line.

HeaderName(r): <S:0..64>

Name of SIP header field to match (e.g. Event).

HeaderPattern(r): <S:0..128>

Regular expression to match against the specified SIP header field.

Authenticated(r): <On/Off>

Whether to forward authenticated requests. **On**: only forward requests along route if incoming message has been authenticated. **Off**: always forward messages that match this route.

Default: Off

Address(r): <S:0..39>

Specifies the IP address of the next hop for this route, where matching SIP requests will be forwarded.

Port(r): <1..65534>

Specifies the port on the next hop for this route to which matching SIP requests will be routed.

Default: 5060

Transport(r): <UDP/TCP/TLS>

Determines which transport type will be used for SIP messages forwarded along this route.

Tag(r): <S:0..64>

Tag value specified by external applications to identify routes that they create.

Example: **xCommand SIPRouteAdd Method: "SUBSCRIBE" RequestLinePattern: ".*@(%localdomains%|ip%)" HeaderName: "Event" HeaderPattern: "(my-event-package)(.*)" Authenticated: On Address: "127.0.0.1" Port: 22400 Transport: TCP Tag: "Tag1"**

SIPRouteDelete

Deletes an existing SIP route, identified either by the specified index or tag.

Note: this command is intended for developer use only.

SipRouteId: <1..20>

The index of the SIP route to be deleted.

Tag: <S:0..64>

Tag value specified by external applications to uniquely identify routes that they create.

Example: **xCommand SIPRouteDelete SipRouteId: Tag: "Tag1"**

SubZoneAdd

Adds and configures a new subzone.

SubZoneName(r): <S: 1, 50>

Assigns a name to this subzone.

TotalMode: <Unlimited/Limited/NoBandwidth>

Determines whether this subzone has a limit on the total bandwidth of calls being used by its endpoints at any one time. NoBandwidth: no bandwidth available. No calls can be made to, from, or within this subzone.

Default: Unlimited.

Total: <1..100000000>

Sets the total bandwidth limit (in kbps) of this subzone (applies only if the mode is set to Limited). Default: 500000.

PerCallInterMode: <Unlimited/Limited/NoBandwidth>

Sets bandwidth limits for any one call to or from an endpoint in this subzone. NoBandwidth: no bandwidth available. No calls can be made to or from this subzone.

Default: Unlimited.

PerCallInter: <1..100000000>

Specifies the bandwidth limit (in kbps) on any one call to or from an endpoint in this subzone (applies only if the mode is set to Limited).

Default: 1920.

PerCallIntraMode: <Unlimited/Limited/NoBandwidth>

Sets bandwidth limits for any one call between two endpoints within this subzone. NoBandwidth: no bandwidth available. No calls can be made within this subzone.

Default: Unlimited.

PerCallIntra: <1..100000000>

Specifies the bandwidth limit (in kbps) for any one call between two endpoints within this subzone (applies only if the mode is set to Limited).

Default: 1920.

Example: **xCommand SubZoneAdd SubZoneName: "BranchOffice" TotalMode: Limited Total: 1024 PerCallInterMode: Limited PerCallInter: 512 PerCallIntraMode: Limited PerCallIntra: 512**

SubZoneDelete

Deletes a subzone.

SubZoneId(r): <1..1000>

The index of the subzone to be deleted.

Example: **xCommand SubZoneDelete SubZoneId: 2**

SubZoneMembershipRuleAdd

Adds and configures a new membership rule.

Name(r): <S: 1..50>

Assigns a name to this membership rule.

Type(r): <Subnet/AliasPatternMatch>

The type of address that applies to this rule. **Subnet**: assigns the device if its IP address falls within the configured IP address subnet. **Alias Pattern Match**: assigns the device if its alias matches the configured pattern.

SubZoneName(r): <S: 1..50>

The subzone to which an endpoint is assigned if its address satisfies this rule.

Description: <S: 0..64>

A free-form description of the membership rule.

Example: **xCommand SubZoneMembershipRuleAdd Name: "Home Workers" Type: Subnet SubZoneName: "Home Workers" Description: "Staff working at home"**

SubZoneMembershipRuleDelete

Deletes a membership rule.

SubZoneMembershipRuleId(r): <1..3000>

The index of the membership rule to be deleted.

Example: **xCommand SubZoneMembershipRuleDelete SubZoneMembershipRuleId: 1**

TransformAdd

Adds and configures a new transform.

Pattern(r): <S: 1, 60>

Specifies the pattern against which the alias is compared.

Type: <Exact/Prefix/Suffix/Regex>

How the pattern string must match the alias for the transform to be applied. **Exact**: the entire string must exactly match the alias character for character. **Prefix**: the string must appear at the beginning of the alias. **Suffix**: the string must appear at the end of the alias. **Regex**: the string is treated as a regular expression.

Default: Prefix

Behavior: <Strip/Replace/AddPrefix/AddSuffix>

How the alias is modified. **Strip**: removes the matching prefix or suffix from the alias. **Replace**: substitutes the matching part of the alias with the text in the replace string. **AddPrefix**: prepends the replace string to the alias. **AddSuffix**: appends the replace string to the alias.

Default: Strip

Replace: <S: 0..60>

The text string to use in conjunction with the selected Pattern behavior.

Priority: <1..65534>

Assigns a priority to the specified transform. Transforms are compared with incoming aliases in order of priority, and the priority must be unique for each transform.

Default: 1

Description: <S: 0..64>

A free-form description of the transform.

State: <Enabled/Disabled>

Indicates if the transform is enabled or disabled. Disabled transforms are ignored.

Default: Enabled

Example: **xCommand TransformAdd Pattern: "example.net" Type: suffix Behavior: replace Replace: "example.com" Priority: 3 Description: "Change example.net to example.com" State: Enabled**

TransformDelete

Deletes a transform.

TransformId(r): <1..100>

The index of the transform to be deleted.

Example: **xCommand TransformDelete TransformId: 2**

UserLoginGroupAdd

Creates a new user login group.

Name(r): <S: 0..128>

Defines the name of a user group that determines which access rights members of the group have after they have been successfully authenticated to use the VCS.

Access(r): <None/ReadWrite>

Defines the access level for members of the specified user group. **None**: no access allowed. **ReadWrite**: configuration can be viewed and changed.

Default: ReadWrite

Example: **xCommand UserLoginGroupAdd Name: "FindMeAccounts" Access: ReadWrite**

Command reference - xCommand

UserLoginGroupDelete

Deletes a user login group.

UserLoginGroupId(r): <1..15>

The index of the user login group to be deleted.

Example: **xCommand UserLoginGroupDelete UserLoginGroupId: 1**

WarningAcknowledge

Acknowledges an existing warning.

Note: this command is intended for developer use only.

WarningID(r): <S:36..36>

The warning ID

Example: **xCommand WarningAcknowledge WarningID: "ab3d63f6-c0bb-4a9c-a121-e683abfedff0"**

WarningLower

Lowers a warning.

Note: this command is intended for developer use only.

WarningID(r): <S:36..36>

The warning ID.

Example: **xCommand WarningLower WarningID: "ab3d63f6-c0bb-4a9c-a121-e683abfedff0"**

WarningRaise

Raises a warning.

Note: this command is intended for developer use only.

WarningID(r): <S:36..36>

The warning ID.

WarningText(r): <S:0..255>

The description of the warning.

Example: **xCommand WarningRaise WarningID: "ab3d63f6-c0bb-4a9c-a121-e683abfedff0" WarningText: "Module foo is malfunctioning"**

ZoneAdd

Adds and configures a new zone.

ZoneName(r): <S: 1, 50>

Assigns a name to this zone.

Type(r): <Neighbor/TraversalClient/TraversalServer/ENUM/DNS>

Determines the nature of the specified zone, in relation to the local VCS. **Neighbor**: the new zone will be a neighbor of the local VCS. **TraversalClient**: there is a firewall between the zones, and the local VCS is a traversal client of the new zone. **TraversalServer**: there is a firewall between the zones and the local VCS is a traversal server for the new zone. **ENUM**: the new zone contains endpoints discoverable by ENUM lookup. **DNS**: the new zone contains endpoints discoverable by DNS lookup.

Example: **xCommand ZoneAdd ZoneName: "UK Sales Office" Type: Neighbor**

Command reference - xCommand

ZoneDelete

Deletes a zone.

ZoneId(r): <1..1000>

The index of the zone to be deleted.

Example: **xCommand ZoneDelete ZoneId: 2**

ZoneList

A diagnostic tool that returns the list of zones (grouped by priority) that would be queried, and any transforms that would be applied, in a search for a given alias. Note that this command does not change any existing system configuration.

Alias(r): <S: 1, 60>

The alias to be searched for.

Example: **xCommand ZoneList Alias: "john.smith@example.com"**

Overview

The **xStatus** group of commands are used to return information about the current status of the VCS. Each **xStatus** element returns information about one or more sub-elements.

The following pages list all the **xStatus** commands currently available on the VCS, and the information that is returned by each.

To obtain information about the existing status on the VCS:

- type **xStatus** to return the current status of all status elements on the VCS.
- type **xStatus <element>** to return the current status for that particular element and all its sub-elements.
- type **xStatus <element> <sub-element>** to return the current status of that group of sub-elements.

To obtain information about the **xStatus** commands:

- type **xStatus ?** to return a list of all elements available under the **xStatus** command.

Command reference - xStatus

Alternates:

Peer [1..6]: {Hidden for Peer [n] when Peer [n] is self}

Status: <Active/Failed/Unknown>

Cause: {Visible if status is Failed} <No response from gatekeeper/DNS resolution failed/Invalid IP address>

Address: <IPv4Addr/IPv6Addr>

Port: <1..65534>

LastStatusChange: <Seconds since boot/Date Time>

Applications:

Presence:

UserAgent:

Status: <Inactive/Initializing/Active/Failed>

Presentity:

Count: <0..2500>

Server:

Publications:

Presentities:

Count: <0..10000>

Max: <0..10000>

Presentity [1..10000]:

URI: <S: 1,255>

Document:

Count: <1..10>

Command reference - xStatus

Subscriptions:

Subscribers:

Count: <0..n>

Max: <0..n>

Subscriber [1..2500]:

URI: <S: 1,255>

Subscription:

Count: <1..100>

Count: <1..2500>

Max: <1..2500>

Expired: <1..2500>

Presentities:

Count: <0..10000>

Max: <0..10000>

Presentity [1..10000]:

URI: <S: 1,255>

Subscriber:

Count: <1..100>

ConferenceFactory:

Status: <Inactive/Initializing/Active/Failed>

NextId: <0.. 4294967295>

External

Status:

Relay:

Registrations:

Count: <1..2500>

Subscriptions:

Count: <1..2500>

Command reference - xStatus

User 1:

Alias: <S: 1,255>

Subscription:

State: <Subscription request sent/Subscription successful/Subscription error response/Failed/Notification received/Active>

Registration:

State: <Registered/Not Registered>

Presence:

OCS:

Machine:

State: <Offline/Available/Undefined>

User:

State: <Undefined/Busy>

VCS:

State: <Offline/Online/In a call/Undefined>

LastUpdate:

Time: <date time>

SecondsSinceLastRefresh: <seconds>

Calls:

Call <1..900>:

SerialNumber: <S: 1,255>

Tag: <S: 1,255>

State: <Connecting/Connected/Disconnecting>

StartTime: <Seconds since boot/Date Time>

Duration: <Time in seconds, precision in seconds>

Legs:

Leg [1..300]:

Protocol: <H323/SIP>

H323: {visible if Protocol = H323}

CallSignalAddress: <IPv4Addr/[IPv6Addr]>:<1..65534>

Aliases:

Alias [1..50]:

Type: <E164/H323Id>

Value: <S: 1,60>

SIP: {visible if Protocol = SIP}

Address: <IPv4Addr/[IPv6Addr]>:<1..65534>

Transport: <UDP/TCP/TLS/undefined>

Aliases:

Alias [1..50]:

Type: <URL>

Value: <S: 1,60>

EncryptionType: <None/DES/AES-128>

CheckCode: <S: 1,60> {visible if Leg = H323 and call is interworked}

Targets:

Target [1..1]:

Type: <E164/H323Id/URL>

Value: <S: 1,60>

BandwidthNode: <S: 1,50 Node name>

Registration:

ID: <1..2500>

SerialNumber: <S: 1,255>

VendorInfo: <S: 1,255>

Sessions:

Session: [1..300:]

Status: <Unknown/Searching/Failed/Cancelled/Completed/Active/Connected>

MediaRouted: <True/False>

CallRouted: <True/False>

Participants:

Leg: <1..300> {2 entries}

Command reference - xStatus

Bandwidth:

Requested: <0..100000000> kbps

Allocated: <0..100000000> kbps

Route:

Zone/Link: <S: 1,50 Node name> {0..150 entries}

Media {visible if MediaRouted = True}

Channels

Channel [1..n]

Type: <AUDIO/VIDEO/DATA/BFCP/H224/UNKNOWN>

Protocol: <S: 1,20> {RTP Payload Type}

Rate: <0.. 4294967295> bps

Packets:

Forwarded:

Total: <0.. 4294967295>

Incoming:

Leg: <1..300>

Outgoing:

Leg: <1..300>

Ethernet [1..2]:

MacAddress: <S: 17>

Speed: <10half/10full/100half/100full/1000full/down>

IPv4:

Address: <IPv4Addr>

SubnetMask: <IPv4Addr>

IPv6:

Address: <IPv6Addr>

Command reference - xStatus

External Manager:

Status: <Inactive/Initializing/Active/Failed>

Cause: {Visible if status is Failed} <Failed to connect to external manager / No response from external manager / Failed to register to external manager / DNS resolution failed >

Address: <IPv4Addr/IPv6Addr >

Protocol: HTTP

URL: <S: 0, 255>

Feedback [1..3]:

Status: <On/Off>

URL: <S: 1,255>

Expression: <S: 1,127> {0..15 entries}

FindMeManager:

Mode: <Off/Local/Remote>

Status: <Active/Inactive/Unknown> {Visible if Remote}

Address: <1..1024> {Visible if Remote}

H323:

Registration:

Status: <Active/Inactive/Failed>

IPv4: {Visible if Status=Active}

Address: <IPv4Addr> {1..2 entries}

IPv6: {Visible if Status=Active}

Address: <IPv6Addr> {1..2 entries}

OutOfResources: <True/False>

CallSignaling:

Status: <Active/Inactive/Failed>

IPv4: {Visible if Status=Active}

Address: <IPv4Addr> {1..2 entries}

IPv6: {Visible if Status=Active}

Address: <IPv6Addr> {1..2 entries}

Command reference - xStatus

CISCO TELEPRESENCE VIDEO COMMUNICATION SERVER
ADMINISTRATOR GUIDE

Assent:

CallSignaling:

Status: <Active/Inactive/Failed>

IPv4: {Visible if Status=Active}

Address: <IPv4Addr> {1..2 entries}

IPv6: {Visible if Status=Active}

Address: <IPv6Addr> {1..2 entries}

H46018:

CallSignaling:

Status: <Active/Inactive/Failed>

IPv4: {Visible if Status=Active}

Address: <IPv4Addr> {1..2 entries}

IPv6: {Visible if Status=Active}

Address: <IPv6Addr> {1..2 entries}

IP:

Protocol: <IPv4/IPv6/Both>

IPv4:

Gateway: <IPv4Addr>

IPv6:

Gateway: <IPv6Addr>

DNS:

Server [1-5]:

Address: <IPv4Addr/IPv6Addr>

Domain: <S: 0, 128>

Command reference - xStatus

LDAP:

Status: <Inactive/Initializing/Active/Failed>

Cause: {Visible if status is Failed} <Failed to connect to LDAP server / The LDAP server does not support TLS. / Failed to establish a TLS connection to the LDAP server. Please check that the LDAP server certificate is signed by a CA, and that CA is included on the CA certificate installed on the VCS. / Failed to authenticate with LDAP server / A valid CA certificate for the LDAP database has not been uploaded; this is required for connections via TLS / No server address configured>

Address: <IPv4Addr/IPv6Addr>

Port: <1..65534>

Links:

Link [1..100]:

Name: <S: 1,50 Link name>

Bandwidth:

LocalUsage: <0..100000000>

ClusterUsage: <0..100000000>

Calls:

Call [0..900]: {0..900 entries}

CallSerialNumber: <S: 1,255>

Loggers

Logger [1..6]

Module:

TraceLevel:

NTP:

Status: <Inactive/Initializing/Active/Failed>

Cause: {Visible if status is Failed} <No response from NTP server/ DNS resolution failed>

Address: <IPv4Addr/IPv6Addr>

Port: <1..65534>

Last Update: <date-time>

Last Correction: <Time in seconds, precision in seconds>

Command reference - xStatus

Options:

Option [1-64]:

Key: <S: 1, 90>

Description: <S: 1, 128>

Pipes:

Pipe [1..1000]:

Name: <S: 1,50 Pipe name>

Bandwidth:

LocalUsage: <0..100000000>

ClusterUsage: <0..100000000>

Calls:

Call [0..900]: {0..900 entries}

CallID: <S: 1,255>

Registrations:

Registration [1..3750]:

Protocol: <H323/SIP>

Node: <S: 1,50 Node name>

SerialNumber: <S: 1,255>

CreationTime: <Date Time>

Duration: <Time in seconds, precision in seconds>

SecondsSinceLastRefresh: <1..65534> {Visible if Protocol is SIP}

SecondsToExpiry <1..65534> {Visible if Protocol is SIP}

VendorInfo: <S: 1,255>

H323: {Visible if Protocol is H323}

Type: <Endpoint/MCU/Gateway/Gatekeeper/MCUGateway>:

CallSignalAddresses:

Address: <IPv4Addr/[IPv6Addr]>:<1..65534>

Command reference - xStatus

RASAddresses:

Address: <IPv4Addr/[IPv6Addr]>:<1..65534>

Apparent: <IPv4Addr/[IPv6Addr]>:<1..65534>

Prefix: <S: 1,20> {0..50 entries}

Aliases:

Alias [1..50]:

Type: <E164/H323Id/URL/Email/GW Prefix/MCU Prefix/Prefix/Suffix/IPAddress>

Origin: <Endpoint/LDAP/Combined>

Value: <S: 1,60>

Traversal: <Assent/H46018> {Visible for Traversal registration}

SIP: {Visible if Protocol is SIP}

AOR: <S: 1,128>

Contact: <S: 1,255>

Path:

URI [1..10]: <S: 1,255>

ResourceUsage:

Calls:

Traversal:

Current: <0..150>

Max: <0..150>

Total: <0..4294967295>

NonTraversal:

Current: <0..750>

Max: <0..750>

Total: <0..4294967295>

Registrations:

Current: <0..3750>

Max: <0..3750>

Total: <0..4294967295>

Command reference - xStatus

TURN:

Relays:

Current: <0..1400>

Max: <0..1400>

Total: <0..4294967295>

SIP:

Ethernet [1..2]

IPv4:

UDP:

Status: <Active/Inactive/Failed>

Address: <IPv4Addr>

TCP:

Status: <Active/Inactive/Failed>

Address: <IPv4Addr>

TLS:

Status: <Active/Inactive/Failed>

Address: <IPv4Addr>

IPv6:

UDP:

Status: <Active/Inactive/Failed>

Address: <IPv6Addr>

TCP:

Status: <Active/Inactive/Failed>

Address: <IPv6Addr>

TLS:

Status: <Active/Inactive/Failed>

Address: <IPv6Addr>

Command reference - xStatus

SystemUnit:

Product: TANDBERG VCS

Uptime: <Time in seconds>

SystemTime: <Time not set/date-time>

TimeZone: <GMT or one of 300 other timezones>

LocalTime: <local-date-time>

Software:

Version: X5.1

Build: <Number/Uncontrolled>

Name: "Release"

ReleaseDate: <Date>

ReleaseKey <ReleaseKey>

Configuration:

NonTraversalCalls: <0..500>

TraversalCalls: <0..100>

Registrations: <0..2500>

Expressway: <True/False>

Encryption: <True/False>

Interworking: <True/False>

FindMe: <True/False>

DeviceProvisioning: <True/False>

DualNetworkInterfaces: <True/False>

AdvancedAccountSecurity: <True/False>

StarterPack: <True/False>

Hardware:

Version: 1.0

SerialNumber: <hardware serial number>

Command reference - xStatus

TURN:

Server:

Status: <Active/Inactive>

Interface [1..2]:

Address: <IPv4Addr/IPv6Addr>

Relays:

Count: <0..1400>

Relay [1..1400]:

Address: <IPv4Addr/IPv6Addr>

Client:

Address: <IPv4Addr/IPv6Addr>

CreationTime: <Date Time>

ExpireTime: <Date Time>

Permissions:

Count: <0..65535>

Permission [0..65535]:

Address: <IPv4Addr/IPv6Addr>

CreationTime: <Date Time>

ExpireTime: <Date Time>

Channels:

Count: <0..65535>

Channel [0..65535]:

ID: <1..65535>

Peer:

Address: <IPv4Addr/IPv6Addr>

CreationTime: <Date Time>

ExpireTime: <Date Time>

Counters:

Received:

Requests:

Total: <0..65535>

Allocate: <0..65535>

Refresh: <0..65535>

Permission: <0..65535>

ChannelBind: <0..65535>

Sent:

Responses:

Total: <0..65535>

Allocate: <0..65535>

Refresh: <0..65535>

Permission: <0..65535>

ChannelBind: <0..65535>

Errors:

Total: <0..65535>

Allocate: <0..65535>

Refresh: <0..65535>

Permission: <0..65535>

ChannelBind: <0..65535>

Media:

Forwarded:

From: <0..65535>

To: <0..65535>

Errors:

From:

NoChannel: <0..65535>

NoPermission: <0..65535>

InvalidType: <0..65535>

FilterFailure: <0..65535>

To:

Command reference - xStatus

NoChannel: <0..65535>

NoPermission: <0..65535>

InvalidType: <0..65535>

FilterFailure: <0..65535>

Warnings:

Warning [1..n]:

ID: <S: 36..36>

Reason: <S: 0..255>

State: <Acknowledged/Unacknowledged>

Zones:

DefaultZone:

Name: "DefaultZone"

Bandwidth:

LocalUsage: <0..100000000>

ClusterUsage: <0..100000000>

Calls: {Section visible only if there are calls }

Call [0..900]: {0..900 entries}

CallId: <S: 1,255>

LocalZone:

DefaultSubZone:

Name: "DefaultSubZone"

Bandwidth:

LocalUsage: <0..100000000>

ClusterUsage: <0..100000000>

Registrations: {0..3750 entries } {Section visible only if there are registrations}

Registration: <1..3750>

SerialNumber: <S: 1,255>

Command reference - xStatus

Calls: {Section visible only if there are calls}

Call [0..900]: {0..900 entries}

CallId: <S: 1,255>

TraversalSubZone:

Name: "TraversalSubZone"

Bandwidth:

LocalUsage: <0..100000000>

ClusterUsage: <0..100000000>

Calls: {Section visible only if there are calls }

Call [0..900]: {0..900 entries}

CallId: <S: 1,255>

ClusterSubZone:

Name: "ClusterSubZone"

Bandwidth:

LocalUsage: <0..100000000>

ClusterUsage: <0..100000000>

Calls: {Section visible only if there are calls }

Call [0..900]: {0..900 entries}

CallId: <S: 1,255>

SubZone: [0..100]

Name: <S: 1,50 Node name>

Bandwidth:

LocalUsage: <0..100000000>

ClusterUsage: <0..100000000>

Registrations: {0..3750 entries} {Section visible only if there are registrations }

Registration: <1..3750>

SerialNumber: <S: 1,255>

Calls: {Section visible only if there are calls}

Call [0..900]: {0..900 entries}

CallId: <S: 1,255>

Command reference - xStatus

Searches:

Current:

Total:

Dropped:

Zone [1..1000]:

Name: <S: 1,50 Node name>

Bandwidth:

LocalUsage: <0..100000000>

ClusterUsage: <0..100000000>

Status: <Active/Failed/Warning>

Cause: {Visible if status is Failed or Warning} <System unreachable/ Systems unreachable>

Type: <Neighbor/TraversalClient/TraversalServer/ENUM/DNS>

Neighbor: {Visible if Type is Neighbor}

Peer [1..6]:

H323: {Visible if H323 Mode=On for Zone}

Status: <Unknown/Active/Failed>

Cause: {Visible if Status is Failed} <No response from gatekeeper/DNS resolution failed/Invalid IP address>

Address: <IPv4Addr/IPv6Addr> {One Address line per address from DNS lookup}

Port: <1..65534>

LastStatusChange: <Time not set/Date Time>

SIP: {Visible if SIP Mode=On for Zone}

Status: <Unknown/Active/Failed>

Cause: {Visible if Status is Failed} <No response from gatekeeper/DNS resolution failed/Invalid IP address>

Address: <IPv4Addr/IPv6Addr> {One Address line per address from DNS lookup}

Port: <1..65534>

LastStatusChange: <Time not set/Date Time>

TraversalClient: {Visible if Type is TraversalClient}

Peer [1..6]:

H323: {Visible if H323 Mode=On for Zone}

Status: <Unknown/Active/Failed>

Cause: {Visible if Status is Failed} <No response from gatekeeper/DNS resolution failed/Invalid alias/Authentication Failed/Invalid IP address>

Command reference - xStatus

Address: <IPv4Addr/IPv6Addr> {One Address line per address from DNS lookup}
Port: <1..65534>
LastStatusChange: <Time not set/Date Time>
SIP: {Visible if SIP Mode=On for Zone}
Status: <Unknown/Active/Failed>
Cause: {Visible if Status is Failed} <No response from neighbor/ DNS resolution failed>
Address: <IPv4Addr/IPv6Addr> {One Address line per address from DNS lookup}
Port: <1..65534>
LastStatusChange: <Time not set/Date Time>

TraversalServer: {Visible if Type is TraversalServer}

SIP:

Port: <Active/Inactive>

H323:

Port: <Active/Inactive>

Peer [1..6]:

H323: {Visible if H323 Mode=On for Zone}

Status: Active

Address: <IPv4Addr/IPv6Addr> {One Address line per address from DNS lookup}

Port: <1..65534>

LastStatusChange: <Time not set/Date Time>

SIP: {Visible if SIP Mode=On for Zone}

Status: Active

Address: <IPv4Addr/IPv6Addr> {One Address line per address from DNS lookup}

Port: <1..65534>

LastStatusChange: <Time not set/Date Time>

Calls: {0..900 entries}

Call [0..900]:

CallID: <S: 1,255>

Overview

Warnings occur when an event or configuration change has taken place on the VCS that requires some manual administrator intervention, such as a restart. Warnings may also be raised for hardware and environmental issues such as faulty disks and fans or high temperatures. The **Warnings** page (**Status > Warnings**) provides a list of all the warnings currently in place on your system (and, where applicable, their proposed resolution).

When there are unacknowledged warnings in place on the VCS, a warning icon  appears at the top right of all pages. The following table lists the warnings that can be raised on the VCS.

Warnings list

Warning message	Resolution
An Ethernet interface speed setting has been negotiated to a value other than 1000Mb/s full duplex or 100Mb/s full duplex. This may result in packet loss over your network	Configure Ethernet parameters
Capacity warning: the number of concurrent non-traversal calls has approached the licensed limit	Contact your Cisco representative
Capacity warning: the number of concurrent traversal calls has approached the licensed limit	Contact your Cisco representative
Capacity warning: the number of concurrent Registrations has approached the licensed limit	Contact your Cisco representative
Cluster name not configured: if FindMe or clustering are in use a cluster name must be defined	Configure the cluster name
Cluster replication error: automatic replication of configuration has been temporarily disabled because an upgrade is in progress	Wait until the upgrade has completed
Cluster replication error: cannot find master or this peer's configuration file, manual synchronization of configuration is required	View cluster replication troubleshooting instructions
Cluster replication error: configuration master ID is inconsistent, manual synchronization of configuration is required	View cluster replication troubleshooting instructions
Cluster replication error: NTP server is not configured	Configure an NTP server
Cluster replication error: the local VCS does not appear in the list of peers	Check the list of peers for this cluster
Cluster replication error: the Master peer is unreachable	Check the list of peers for this cluster
Cluster replication error: the NTP server is unreachable	Reconfigure the NTP server
Cluster replication error: there was an error during automatic replication of configuration	View cluster replication troubleshooting instructions
Cluster replication error: this peer's configuration conflicts with the master's configuration, manual synchronization of configuration is required	View cluster replication troubleshooting instructions
Configuration conflict: a domain is in use by the OCS Relay application that has not been configured on the VCS	Reconfigure the OCS Relay or view and edit the VCS SIP domains
Configuration conflict: the FindMe mode is set to "On" or "Third Party Manager" but the FindMe option key has been deleted	Reconfigure FindMe mode or reinstall the option key
Configuration warning: expected default link between the Default Subzone and the Cluster Subzone is missing	Configure default links
Configuration warning: expected default link between the Default Subzone and the Default Zone is missing	Configure default links
Configuration warning: expected default link between the Default Subzone and the Traversal Subzone is missing	Configure default links

Warnings list

Warning message	Resolution
Configuration warning: expected default link between the Traversal Subzone and the Default Zone is missing	Configure default links
Configuration warning: H.323 and SIP modes are set to Off; one or both of them should be enabled	Configure H.323 and/or SIP modes
Configuration conflict: H323-SIP Protocol Interworking mode is set to "Registered only" but the H323-SIP Interworking Gateway option key has been deleted	Reconfigure Interworking mode or reinstall the option key
Configuration warning: IP protocol is set to both IPv4 and IPv6, but the VCS does not have an IPv4 gateway defined	Configure IP settings
Configuration warning: IP protocol is set to both IPv4 and IPv6, but the VCS does not have an IPv6 gateway defined	Configure IP settings
Configuration warning: IP protocol is set to both IPv4 and IPv6, but the VCS does not have any IPv4 addresses defined	Configure IP settings
Configuration warning: IP protocol is set to both IPv4 and IPv6, but the VCS does not have any IPv6 addresses defined	Configure IP settings
CRL checking required: your login account LDAP server configuration is recommended to have certificate revocation list (CRL) checking set to "All" when in advanced account security mode	Configure login account LDAP server
Encryption required: your login account LDAP server configuration is recommended to have encryption set to "TLS" when in advanced account security mode	Configure login account LDAP server
HTTPS client certificate checking: client certificate checking mode has changed, however a restart is required for this to take effect	Restart the VCS
HTTPS client certificate validation: you are recommended to enable client side certificate validation for HTTPS connections when in advanced account security mode	Configure HTTPS client certificate validation
Invalid clustering configuration: H.323 mode must be turned On - clustering uses H.323 communications between peers	Configure H.323 mode
LDAP Configuration required: a valid Server Address, Server Port, BindDN and BaseDN are required if remote login authentication is used for administrator or FindMe users	Configure LDAP parameters
NTP server problem: the VCS is unable to determine the correct time and date	Check the time configuration
Port conflict: one or more ports have been configured for use by more than one service	Review the port configuration
Restart required: Ethernet configuration has been changed, however a restart is required for this to take effect	Restart the VCS
Restart required: HTTP service has been changed, however a restart is required for this to take effect	Restart the VCS
Restart required: HTTPS service has been changed, however a restart is required for this to take effect	Restart the VCS
Restart required: IP configuration has been changed, however a restart is required for this to take effect	Restart the VCS
Restart required: port configuration has been changed, however a restart is required for this to take effect	Restart the VCS
Restart required: QoS settings have been changed, however a restart is required for this to take effect	Restart the VCS
Restart required: SNMP service has been changed, however a restart is required for this to take effect	Restart the VCS
Restart required: SSH service has been changed, however a restart is required for this to take effect	Restart the VCS

Warnings list

Warning message	Resolution
Restart required: Telnet service has been changed, however a restart is required for this to take effect	Restart the VCS
Restart required: the advanced account security mode has changed, however a restart is required for this to take effect	Restart the VCS
Restart required: the Dual Network Interfaces option key has been changed, however a restart is required for this to take effect	Restart the VCS
Restart required: the TMS Agent service has stopped unexpectedly. If the problem persists, contact your TANDBERG support representative	Restart the VCS
Security alert: one or more administrators has a password that does not meet strictness requirements	View and edit administrator accounts
Security alert: the admin user has the default password set	Change the admin password
Security alert: the root user has the default password set	Change the root password
Security alert: the SSH service is using the default key	Replace the default SSH key
Security alert: the TMS Agent database has the default LDAP password set	Change the TMS Agent LDAP password
Security alert: the TMS Agent database has the default replication password set	Change the TMS Agent replication password
SNMP enabled: you are recommended to disable SNMP when in advanced account security mode	Configure SNMP mode
Time out period required: a non-zero system session time out period is required when in advanced account security mode	Configure session time out period
TURN relays installed: TURN services are only available on VCS Expressway; TURN option key ignored	Add/Remove option keys
TURN relay licenses required: TURN services are enabled but no TURN relay license option keys are installed	Add option key or disable TURN services

Reference	Title	Link
1	ITU Specification: H.235 Security and encryption for H-Series (H.323 and other H.245-based) multimedia terminals	http://www.itu.int/rec/T-REC-H.235/en
2	ITU Specification: H.350 Directory services architecture for multimedia conferencing	http://www.itu.int/rec/T-REC-H.350/en
3	RFC 2782: A DNS RR for specifying the location of services (DNS SRV)	http://www.ietf.org/rfc/rfc2782.txt
4	RFC 3164: The BSD syslog Protocol	http://www.ietf.org/rfc/rfc3164.txt
5	RFC 3880: Call Processing Language (CPL): A Language for User Control of Internet Telephony Services	http://www.ietf.org/rfc/rfc3880.txt
6	DNS and BIND Fourth Edition, Albitz and Liu, O'Reilly and Associates, ISBN: 0-596-00158-4	
7	RFC 2915: The Naming Authority Pointer (NAPTR) DNS Resource Record	http://www.ietf.org/rfc/rfc2915.txt
8	RFC 3761: The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)	http://www.ietf.org/rfc/rfc3761.txt
9	Mastering Regular Expressions, Jeffrey E.F. Friedl, O'Reilly and Associates, ISBN: 1-56592-257-3	
10	RFC 3327: Session Initiation Protocol (SIP) Extension Header Field for Registering Non-Adjacent Contacts	http://www.ietf.org/rfc/rfc3327.txt
11	Session Traversal Utilities for (NAT) (STUN)	http://tools.ietf.org/html/rfc5389
12	Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)	http://tools.ietf.org/html/draft-ietf-behave-turn-16
13	RFC 4787: Network Address Translation (NAT) Behavioral Requirements for Unicast UDP	http://www.ietf.org/rfc/rfc4787.txt
14	RFC 4028: Session Timers in the Session Initiation Protocol (SIP)	http://www.ietf.org/rfc/rfc4028.txt
15	ITU Specification: H.323: Packet-based multimedia communications systems	http://www.itu.int/rec/T-REC-H.323/en
16	RFC 3263: Session Initiation Protocol (SIP): Locating SIP Servers	http://www.ietf.org/rfc/rfc3263.txt
17	RFC 3550: RTP: A Transport Protocol for Real-Time Applications	http://www.ietf.org/rfc/rfc3550.txt
18	RFC 791: Internet Protocol	http://www.ietf.org/rfc/rfc791.txt
19	RFC 2460: Internet Protocol, Version 6 (IPv6) Specification	http://www.ietf.org/rfc/rfc2460.txt
20	RFC 3261: SIP: Session Initiation Protocol	http://www.ietf.org/rfc/rfc3261.txt
21	RFC 3489: STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)	http://www.ietf.org/rfc/rfc3489.txt
22	XML and Writing CPL for TANDBERG Infrastructure products Rev 1.2	www.tandberg.com/support/documentation.php
23	Management Information Base for Network Management of TCP/IP-based internets: MIB-II	http://www.ietf.org/rfc/rfc1213.txt
24	Cisco VCS Deployment Guide - Microsoft OCS 2007 (R1 and R2) and Cisco VCS Control (document no. D14269)	www.tandberg.com/support/documentation.php
25	Cisco VCS Multiway Deployment Guide (document number D14366)	www.tandberg.com/support/documentation.php
26	Provisioning Deployment Guide (document number D14368)	www.tandberg.com/support/documentation.php
27	Cisco VCS Deployment Guide - Cluster creation and maintenance (document number D14367)	www.tandberg.com/support/documentation.php

Bibliography

Reference	Title	Link
28	Cisco VCS Getting Started Guide (document number D14350)	www.tandberg.com/support/documentation.php
29	Cisco VCS Deployment Guide - FindMe (document number D14525)	www.tandberg.com/support/documentation.php
30	Cisco VCS Deployment Guide - Authenticating Cisco VCS accounts using LDAP (document number D14526)	www.tandberg.com/support/documentation.php
31	Cisco VCS Deployment Guide - ENUM dialing on Cisco VCS (document number D14465)	www.tandberg.com/support/documentation.php
32	Cisco VCS Deployment Guide - Certificate Creation and use with Cisco VCS (document number D14548)	www.tandberg.com/support/documentation.php
33	Cisco VCS Deployment Guide - Basic Configuration - Single Cisco VCS Control (document number D14524)	www.tandberg.com/support/documentation.php
34	Cisco VCS Deployment Guide - Cisco VCS Expressway Starter Pack (document number D14618)	www.tandberg.com/support/documentation.php
35	RFC 3326: The Reason Header Field for the Session Initiation Protocol (SIP)	http://www.ietf.org/rfc/rfc3326.txt

Term	Definition
A record	A type of DNS record that maps a host name to an IPv4 address.
AAAA record	A type of DNS record that maps a host name to an IPv6 address.
Administrator Policy	See Call Policy
Alias	The name an endpoint uses when registering with the VCS. Other endpoints can then use this name to call it. An endpoint may register with more than one alias.
Alternate	One or more VCSs configured to support the same zone in order to provide redundancy. See also Cluster.
AOR Address of Record	A SIP or SIPS URI that points to a domain with a location service that can map the URI to another URI where the user might be available. Typically, the location service is populated through registrations. An AOR is frequently thought of as the “public address” of the user.
ARQ Admission Request	An endpoint RAS request to make or answer a call.
Assent	Cisco's proprietary protocol for firewall traversal.
Border Controller	A device used to control multimedia networks and firewall traversal.
CA Certificate authority	An organization that validates and signs certificate requests.
Call Policy	In relation to the VCS, the set of rules configured system-wide (either via the web interface or CPL script) that determine the action(s) to be applied to calls matching a given criteria. (Also referred to as Administrator Policy.)
Cisco TMS Cisco TelePresence Management Suite	A Cisco product used for the management of video networks.
Cisco VCS Cisco TelePresence Video Communication Server	A generic term for the Cisco product which acts as a gatekeeper and SIP proxy/server.
Cisco VCS Control	A Cisco VCS whose main function is to act as a gatekeeper, SIP proxy and firewall traversal client. This system is generally located within the firewall.
Cisco VCS Expressway	A Cisco VCS with the same functionality as a Cisco VCS Control that can also act as a firewall traversal server. This is generally located outside the firewall.
CLI Command line interface	A text-based user interface used to access the VCS.
Cluster	A collection of between two and six VCSs that have been configured to work together as a single Local Zone, in order to provide scalability and redundancy.
Conference Factory	An application that allows the VCS to support the Multiway feature. Cisco TelePresence Multiway enables endpoint users to create a conference while in a call even if their endpoint does not have this functionality built in. See the Conference Factory section for more information.
CPL Call Processing Language	An XML-based language for defining call handling. Defined by RFC 3880 [5] .
CRL Certificate revocation list	A list from a CA (certificate authority) of previously signed certificates that it marks as no longer valid.
Default Subzone	A subzone used to represent locally registered endpoints and systems that do not fall within any other existing configured subzones within the Local Zone.

Term	Definition
Default Zone	A non-configurable zone on the VCS used to represent incoming connections from endpoints that are not recognized as belonging to any of the existing configured neighbor, traversal client or traversal server zones.
Device Provisioning	An option key that allows VCS to provision endpoints with configuration information on request and to supply endpoints with phone book information. See the Device Provisioning section for more information.
DiffServ Differentiated Services	A Quality of Service (QoS) mechanism supported by the VCS.
DNS Domain Name System	A distributed database linking domain names to IP addresses.
DNS zone	On the VCS, a zone used to configure access to endpoints located via a DNS lookup.
E.164	An ITU standard for structured telephone numbers. Each telephone number consists of a country code, area code and subscriber number.
ENUM E.164 Number Mapping	A means of mapping E.164 numbers to URIs using DNS. Defined by RFC 3761 [8] .
ENUM zone	On the VCS, a zone used to configure access to endpoints located via ENUM.
External manager	The remote system that is used to manage endpoints and network infrastructure. The Cisco TelePresence Management Suite (Cisco TMS) is an example of an external manager.
External zone	Any zone configured on the local VCS that connects out to a remote system or the internet. Neighbor, traversal server, traversal client, ENUM and DNS zones are all external zones.
Firewall traversal	The act of crossing a firewall or NAT device.
FindMe™	A User Policy feature that allows users to have a single alias on which they can be reached regardless of the endpoint(s) they are currently using.
FQDN Fully Qualified Domain Name	A domain name that specifies the node's position in the DNS tree absolutely, uniquely identifying the system or device. Note that in order to use FQDNs instead of IP addresses when configuring the VCS, you must have at least one DNS server configured.
Gatekeeper	A device used to control H.323 multimedia networks. An example is the TANDBERG Gatekeeper.
Gatekeeper zone	A collection of all the endpoints, gateways and MCUs managed by a single gatekeeper.
H.323	A standard that defines the protocols used for packet-based multimedia communications systems.
HTTP Hypertext Transfer Protocol	A protocol used for communications over the internet.
HTTPS Hypertext Transfer Protocol over Secure Socket Layer	A protocol used for secure communications over the internet, combining HTTP with TLS.
Hop count	The maximum number of gatekeeper or SIP proxy devices (e.g. a VCS) that a message may be forwarded through before it is decided that its intended recipient is not reachable.
ICE Interactive Connectivity Establishment	A collaborative algorithm that works together with STUN services (and other NAT traversal techniques) to allow clients to achieve firewall traversal. This is the emerging traversal standard for use by SIP endpoints (although it could be used for other protocols).

Term	Definition
IETF Internet Engineering Task Force	An organization that defines (via documents such as RFCs) the protocol standards and best practices relating to the design, use and management of the internet.
Interworking	Allowing H.323 systems to connect to SIP systems.
IPv4 Internet Protocol version 4	Version 4 of the Internet Protocol, defined in RFC 791 [18] .
IPv6 Internet Protocol version 6	Version 6 of the Internet Protocol, defined in RFC 2460 [19] .
IRQ Information Request	A request sent to an endpoint requesting information about its status.
LAN Local Area Network	A geographically limited computer network, usually with a high bandwidth throughput.
LDAP Lightweight Directory Access Protocol	A protocol for accessing on-line directories running over TCP/IP.
Link	In relation to the VCS, a connection between two nodes.
Local call	(Also referred to as a non-traversal call.) A call where the signaling but not the media is routed through the local VCS. See the What are traversal calls? section for more information.
Local registration, Locally registered endpoint	A relative term used to refer to any endpoint or system that is registered with the local VCS.
Local VCS	A relative term used to refer to the particular VCS that you are currently administering, as opposed to other VCSs in your network.
Local Zone	A relative term used to refer to the group of endpoints and other systems registered to a particular VCS. If a VCS is part of a cluster, the Local Zone refers to the collection of all endpoints and other systems registered to all peers in that cluster.
LRQ Location Request	A RAS query between gatekeepers to determine the location of an endpoint.
MCU Multipoint Control Unit	A network device that allows multiple endpoints to participate in a video conference.
Microsoft Office Communications Server 2007 Microsoft OCS	Microsoft OCS (Office Communications Server) 2007 is an enterprise real-time communications server, providing the infrastructure to allow instant messaging, presence, audio-video conferencing and web conferencing functionality.
Microsoft Office Communications (MOC) client	The client application released with Microsoft Office Communications Server (OCS). The MOC client can be used for instant messaging, presence, voice and video calls and ad hoc conferences.
NAPTR record	A type of DNS record.
NAT Network Address Translation	Also known as IP masquerading. Rewriting source and destination addresses as the IP packet passes through the NAT device.
Neighbor	An remote system to which the VCS has a connection via a neighbor zone.

Term	Definition
Neighbor zone	On the VCS, a zone used to configure a connection to a remote system with which the local VCS has a non-traversal relationship.
Node	In relation to the VCS, a node is one end of a link. A node can be a local subzone or a zone.
Non-traversal call	(Also referred to as local call.) A call where the signaling but not the media is routed through the local VCS. See the What are traversal calls? section for more information.
NTP Network Time Protocol	A protocol used for synchronizing clocks. Defined in RFC 1305.
OCS Relay	A VCS application that enables interoperability between Microsoft Office Communications Server (OCS) and FindMe. See the OCS Relay section for more information.
Peer	A VCS that has been configured to belong to a cluster.
PEM Privacy-Enhanced Electronic Mail:	An IETF proposal for securing messages using public key cryptography.
Pipe	In relation to the VCS, a means of controlling the bandwidth used on a link.
Proxy, Proxy server	In SIP, an intermediary entity that acts as both a server and a client for the purpose of making requests on behalf of other clients. A proxy server primarily plays the role of routing, which means its job is to ensure that a request is sent to another entity "closer" to the targeted user. Proxies are also useful for enforcing policy (for example, making sure a user is allowed to make a call). A proxy interprets, and, if necessary, rewrites specific parts of a request message before forwarding it. While a proxy can set up calls between SIP endpoints, it does not participate in the call after it is set up.
QoS Quality of Service	Mechanisms that give a network administrator the ability to provide different priorities to an applications' network traffic.
RAS Registration, Admission and Status	A protocol used between H.323 endpoints and a gatekeeper to register and place calls.
Registrar	In SIP, a server that accepts REGISTER requests and places the information it receives in those requests into the location service for the domain it handles. This information is used to advise other SIP Proxies/Registrars where to send calls for that endpoint.
Regex Regular expression	A pattern used to match text strings according to a POSIX-defined syntax.
RFC Request for Comments	A process and result used by the IETF for Internet standards.
RS-232	A commonly used standard for computer serial ports.
RTCP RTP Control Protocol	A control protocol for RTP. Defined by RFC 3550 [17] .
RTP Real-time Transport Protocol	Real time protocol designed for the transmission of voice and video. Defined by RFC 3550 [17] .
SASL Simple Authentication and Security Layer	A framework for authentication and data security in Internet protocols.

Glossary

Term	Definition
SSH Secure Shell	An encrypted protocol used to provide a secure CLI.
SIMPLE Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions	An instant messaging and presence protocol based on SIP.
SIP Session Initiation Protocol	IETF protocol for controlling multimedia communication. Defined by RFC 3261 [20] .
SNMP Simple Network Management Protocol	A protocol used to monitor network devices.
Source alias	The alias present in the “source” field of a message.
SRV record Service record	A type of DNS record. Defined by RFC 2782.
STUN Simple Traversal of UDP through NATs	Firewall NAT traversal for SIP. Defined by RFC 3489 [21] .
Subzone	A segment within a VCS Local Zone.
TCP Transmission Control Protocol	A reliable communication protocol defined by RFC 791 [18] .
Telnet	A network protocol used on the internet or Local Area Networks (LANs).
TLS Transport Layer Security	A protocol that provides secure communications over the internet.
TMS	See Cisco TMS.
TMS Agent	An internal VCS feature used to manage FindMe and Device Provisioning data. See the TMS Agent section for more information.
Transform	In relation to the VCS, the process of changing or replacing the alias being searched for.
Traversal call	Any call where both signaling and media are routed through the local VCS. See the What are traversal calls? section for more information.
Traversal client	A traversal entity on the private side of a firewall. Examples are a VCS Control or TANDBERG Gatekeeper.
Traversal client zone	A zone on a VCS traversal client that has been used to configure a connection to a particular traversal server.
Traversal server	A traversal entity on the public side of a firewall. Examples are the VCS Expressway or a TANDBERG Border Controller.
Traversal server zone	A zone on a VCS Expressway that has been used to configure a connection to a particular traversal client.
Traversal Subzone	A conceptual subzone through which all traversal calls are deemed to pass; used to manage the bandwidth of traversal calls.
Traversal-enabled endpoint	Any endpoint that supports the Assent and/or ITU H.460.18 and H.460.19 standards for firewall traversal. This includes all Cisco TelePresence MXP endpoints.
TURN Traversal Using Relays around NAT	Relay extensions to STUN (Session Traversal Utilities for NAT).

Glossary

Term	Definition
UA User Agent	A SIP device used to make and receive calls.
UDP User Datagram Protocol	A communication protocol defined by RFC 791 [18] . It is less reliable than TCP.
URI Uniform Resource Identifier	A formatted string used to identify a resource, typically on the internet.
User Policy	The set of rules that determine the action(s) to be applied to calls for a particular user or group. The VCS uses FindMe for its User Policy.
VCS	See Cisco VCS.
VCS Control	See Cisco VCS Control.
VCS Expressway	See Cisco VCS Expressway.
Zone	Zones are used on the VCS to define and configure connections to locally registered and external systems and endpoints. The Local Zone refers to all the locally registered endpoints and systems, and consists of configurable subzones. External zones are used to configure connections to external systems with which the VCS has a neighbor, traversal client or traversal server relationship, and to configure the way in which the VCS performs ENUM and DNS searches.

Disclaimer

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks.

Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2010 Cisco Systems, Inc. All rights reserved.

Copyright notice

The product that is covered by this Administrator Guide is protected under copyright, patent, and other intellectual property rights of various jurisdictions.

This product is

Copyright © 2010, Tandberg Telecom UK Limited. All rights reserved.

TANDBERG is now part of Cisco. Tandberg Telecom UK Limited is a wholly owned subsidiary of Cisco Systems, Inc.

This product includes copyrighted software licensed from others. A list of the copyright notices and the terms and conditions of use can be found at:

http://www.tandberg.com/collateral/documentation/User_Manuals/VCS_EULA.pdf

and

http://www.tandberg.com/collateral/documentation/User_Manuals/VCS_Copyrights.pdf.

IMPORTANT: USE OF THIS PRODUCT IS SUBJECT IN ALL CASES TO THE COPYRIGHT RIGHTS AND THE TERMS AND CONDITIONS OF USE REFERRED TO ABOVE. USE OF THIS PRODUCT CONSTITUTES AGREEMENT TO SUCH TERMS AND CONDITIONS.

Intellectual property rights

This Administrator Guide and the product to which it relates contain information that is proprietary to TANDBERG and its licensors. Information regarding the product is found adjacent in the **Copyright notice** and **Patent information** sections.

TANDBERG® is a registered trademark belonging to Tandberg ASA. Other trademarks used in this document are the property of their respective holders.

This Guide may be reproduced in its entirety, including all copyright and intellectual property notices, in limited quantities in connection with the use of this product. Except for the limited exception set forth in the previous sentence, no part of this Guide may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronically, mechanically, by photocopying, or otherwise, without the prior written permission of TANDBERG.

COPYRIGHT © 2010, TANDBERG
TANDBERG is now part of Cisco.

Patent information

This product is covered by one or more of the following patents:

- US7,512,708
- EP1305927
- EP1338127

A complete list of patents is available at:
http://www.tandberg.com/tandberg_pm.jsp.