TANDBERG Gatekeeper
User Guide

Software version N6.0

D13381.09

February 2008

Contents

1. Product Information

1.1. Trademarks and Copyright

Copyright 1993-2008 TANDBERG ASA. All rights reserved.

This document contains information that is proprietary to TANDBERG ASA. No part of this publication 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 ASA. Nationally and internationally recognized trademarks and tradenames are the property of their respective holders and are hereby acknowledged.

Portions of this software are licensed under 3rd party licenses. See the CD accompanying this product for details. 3rd party license information may also be obtained from the Gatekeeper itself -- see the `license` command in section 17.6.4 for details.

1.2. Disclaimer

The information in this document is furnished for informational purposes only, is subject to change without prior notice, and should not be construed as a commitment by TANDBERG ASA.

The information in this document is believed to be accurate and reliable, however TANDBERG ASA assumes no responsibility or liability for any errors or inaccuracies that may appear in this document, nor for any infringements of patents or other rights of third parties resulting from its use. No license is granted under any patents or patent rights of TANDBERG ASA.

COPYRIGHT ©2008, TANDBERG ASA

1.3. Environmental Issues

Thank you for buying a product which contributes to a reduction in pollution, and thereby helps save the environment. Our products reduce the need for travel and transport and thereby reduce pollution. Our products have either none or few consumable parts (chemicals, toner, gas, paper). Our products are low energy consuming products.

1.3.1. TANDBERG's Environmental Policy

Environmental stewardship is important to TANDBERG's culture. As a global company with strong corporate values, TANDBERG is committed to being an environmental leader and embracing technologies that help companies, individuals and communities creatively address environmental challenges.

TANDBERG's environmental objectives are to:

- Develop products that reduce energy consumption, CO emissions, and traffic congestion
- Provide products and services that improve quality of life for our customers
- Produce products that can be recycled or disposed of safely at the end of product life
- Comply with all relevant environmental legislation.

1.3.2. European Environmental Directives

As a manufacturer of electrical and electronic equipment TANDBERG is responsible for compliance with the requirements in the European Directives 2002/96/EC (WEEE) and 2002/95/EC (RoHS).

The primary aim of the WEEE Directive and RoHS Directive is to reduce the impact of disposal of electrical and electronic equipment at end-of-life. The WEEE Directive aims to reduce the amount of WEEE sent for disposal to landfill or incineration by requiring producers to arrange for collection and recycling. The RoHS Directive bans the use of certain heavy metals and brominates flame retardants to reduce the environmental impact of WEEE which is land filled or incinerated.

TANDBERG has implemented necessary process changes to comply with the European RoHS Directive (2002/95/EC) and the European WEEE Directive (2002/96/EC).

1.3.3. Waste Handling

In order to avoid the dissemination of hazardous substances in our environment and to diminish the pressure on natural resources, we encourage you to use the appropriate take-back systems in your area. Those systems will reuse or recycle most of the materials of your end of life equipment in a sound way.

TANDBERG products put on the market after August 2005 are marked with a crossed-out wheelie bin symbol that invites you to use those take-back systems.

Please contact your local supplier, the regional waste administration or http://www.tandberg.net/recycling if you need more information on the collection and recycling system in your area.

1.3.4. Information for Recyclers

As part of compliance with the European WEEE Directive, TANDBERG provides recycling information on request for all types of new equipment put on the market in Europe after August 13th 2005.

Please contact TANDBERG at recycling@tandberg.net and provide the following details for the product for which you would like to receive recycling information:

- Model number of TANDBERG product
- Your company's name
- Contact name
- Address
- Telephone number
- E-mail address

1.3.5. Digital User Guides

TANDBERG is pleased to announce that we have replaced the printed versions of our User Guides with a digital CD version. Instead of a range of different user manuals, there is now one CD -- which can be used with all TANDBERG products -- in a variety of languages. The environmental benefits of this are significant. The CDs are recyclable and the savings on paper are huge. A simple web-based search feature helps you directly access the information you need. The contents of the CD can still be printed locally, whenever needed.

1.4. Operator Safety Summary

For your protection please read these safety instructions completely before you connect the equipment to the power source. Carefully observe all warnings, precautions and instructions both on the apparatus and in these operating instructions.

Keep this manual for future reference.

1.4.1. Water and Moisture

- Do not operate the apparatus under or near water - for example near a bathtub, kitchen sink, or laundry tub, in a wet basement, near a swimming pool or in other areas with high humidity.
- Never install jacks for communication cables in wet locations unless the jack is specifically designed for wet locations.
- Do not touch the product with wet hands.

1.4.2. Cleaning

- Unplug the apparatus from communication lines, mains power-outlet or any power source before cleaning or polishing. Do not use liquid cleaners or aerosol cleaners. Use a lint-free cloth lightly moistened with water for cleaning the exterior of the apparatus.
- Unplug the apparatus from communication lines before cleaning or polishing. Do not use liquid cleaners or aerosol cleaners. Use a lint-free cloth lightly moistened with water for cleaning the exterior of the apparatus.

1.4.3. Ventilation

- Do not block any of the ventilation openings of the apparatus. Never cover the slots and openings with a cloth or other material. Never install the apparatus near heat sources such as radiators, heat registers, stoves, or other apparatus (including amplifiers) that produce heat.
- Do not place the product in direct sunlight or close to a surface directly heated by the sun.

1.4.4. Lightning

- Never use this apparatus, or connect/disconnect communication cables or power cables during lightning storms.

1.4.5. Dust

- Do not operate the apparatus in areas with high concentration of dust.

1.4.6. Vibration

- Do not operate the apparatus in areas with vibration or place it on an unstable surface.

1.4.7. Power connection and Hazardous voltage

- The product may have hazardous voltage inside. Never attempt to open this product, or any peripherals connected to the product, where this action requires a tool.

- This product should always be powered from an earthed power outlet.

- Never connect attached power supply cord to other products.

- If any parts of the product have visual damage, do not attempt to connect mains power (or any other power source) before consulting service personnel.

- The plug connecting the power cord to the product/power supply serves as the main disconnect device for this equipment. The power cord must always be easily accessible.

- Route the power cord so as to avoid it being walked on or pinched by items placed upon or against it. Pay particular attention to the plugs, receptacles and the point where the cord exits from the apparatus.

- Do not tug the power cord.

- If the provided plug does not fit into your outlet, consult an electrician. Never install cables, or any peripherals, without first unplugging the device from its power source.

1.4.8. Servicing

- 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.

- Unplug the apparatus from its power source and refer servicing to qualified personnel under the following conditions:

  - If the power cord or plug is damaged or frayed.
  - If liquid has been spilled into the apparatus.
  - If objects have fallen into the apparatus.
  - If the apparatus has been exposed to rain or moisture.
  - If the apparatus has been subjected to excessive shock by being dropped.
  - If the cabinet has been damaged.
  - If the apparatus seems to be overheated.
  - If the apparatus emits smoke or abnormal odor.
  - If the apparatus fails to operate in accordance with the operating instructions.

1.4.9. Accessories

- Use only accessories specified by the manufacturer, or sold with the apparatus.

2. Introduction

This User Manual is provided to help you make the best use of your TANDBERG Gatekeeper.

2.1. Main Features

The main features of the TANDBERG Gatekeeper are:

- IPv4 and IPv6 support.
- Supports up to 2500 registered endpoints.
- Supports up to 100 neighboring zones.
- Flexible zone configuration with prefix and suffix support.
- URI and ENUM dialing with DNS enabling global connectivity.
- Secure firewall traversal of any firewall or NAT when used in conjunction with a TANDBERG Border Controller.
- Up to 500 concurrent calls.
- Up to 100 traversal calls in conjunction with a TANDBERG Border Controller.
- Can be used to control the amount of bandwidth used both within the Gatekeeper zone and to neighboring Border Controllers and Gatekeepers.
- Can limit total bandwidth usage and set maximum per call bandwidth usage with automatic downspeeding if call exceeds per-call maximum.
- Can be managed with TANDBERG Management Suite 11.0 or newer, or as a standalone system with RS-232, Telnet, SSH, HTTP and HTTPS.
- Embedded setup wizard on serial port for initial configuration.

Note: features may vary depending on software package.

2.2. Hardware Overview

On the front of the Gatekeeper (see Figure 1) there are:

- three LAN interfaces
- a serial port (Data 1)
- a Light Emitting Diode (LED) showing the power status of the system.

The LAN 1 interface is used for connecting the system to your network. LAN interface 2 and 3 are disabled.

The serial port (Data 1) is for connection to a PC.

The LED, when lit, indicates that power is on.

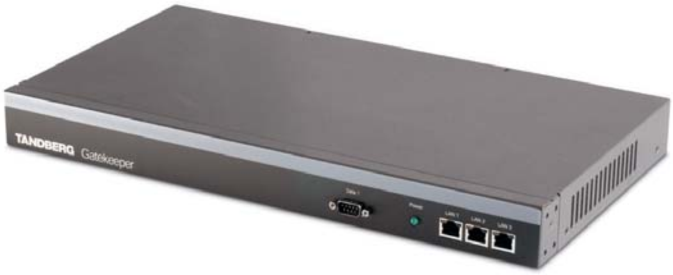


Figure 1: Front panel of Gatekeeper

On the back of the Gatekeeper (see Figure 2) there are:

- a power connector
- a power switch
- a serial port (Data 2) for connecting to a PC.

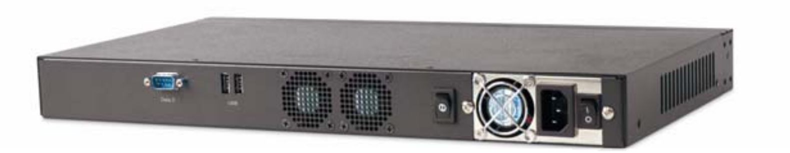


Figure 2: Rear panel of Gatekeeper

3. Installation

3.1. Precautions

- Never install communication equipment during a lightning storm.
- Never install jacks for communication cables in wet locations unless the jack is specifically designed for wet locations.
- Never touch uninstalled communication wires or terminals unless the communication line has been disconnected at the network interface.
- Use caution when installing or modifying communication lines.
- Avoid using communication equipment (other than a cordless type) during an electrical storm.
- There may be a remote risk of electrical shock from lightning.
- Do not use communication equipment to report a gas leak in the vicinity of the leak.
- The socket outlet shall be installed near to the equipment and shall be easily accessible.
- Never install cables without first switching the power OFF.
- This product complies with directives: LVD 73/23/EC and EMC 89/366/EEC.
- Power must be switched off before power supplies can be removed from or installed into the unit.

3.2. Preparing the Installation Site

- Make sure that the Gatekeeper is accessible and that all cables can be easily connected.
- For ventilation: Leave a space of at least 10cm (4 inches) behind the Gatekeeper's rear and 5cm (2 inches) on the sides.
- The room in which you install the Gatekeeper should have an ambient temperature between 0C and 35C (32F and 95F) and between 10% and 90% non-condensing relative humidity.
- Do not place hot objects directly on top of or directly beneath the Gatekeeper.
- Use a grounded AC power outlet for the Gatekeeper.

3.3. Unpacking

The TANDBERG Gatekeeper is delivered in a special shipping box which should contain the following components:

- Gatekeeper unit
- Installation sheet
- User manual and other documentation on CD
- Rack-ears and screws
- Kit with 4 rubber feet
- Cables:
    - Power cables
    - One Ethernet cable
    - One null-modem RS-232 cable

3.4. Mounting

The Gatekeeper comes with brackets for mounting in standard 19" racks.

Before starting the rack mounting, please make sure the TANDBERG Gatekeeper is placed securely on a hard, flat surface.

1. Disconnect the AC power cable.
2. Make sure that the mounting space is according to the Installation site preparations in section 3.2.
3. Attach the brackets to the chassis on both sides of the unit.
4. Insert the unit into a 19" rack, and secure it with screws.

3.5. Connecting the Cables

3.5.1. Power cable

Connect the system power cable to an electrical distribution socket.

3.5.2. LAN cable

Connect a LAN cable from the LAN 1 connector on the front of the unit to your network.

3.5.3. Null-modem RS-232 cable

Connect the supplied null-modem RS-232 cable between the Border Controller's Data 1 connector and the COM port on a PC.

3.6. Switching on the System

To start the TANDBERG Gatekeeper:

1. Ensure the power cable is connected.
2. Ensure the LAN cable is connected.
3. Switch the power switch button on the back of the unit to '1'.

On the front of the chassis you will see the Power LED being lit.

4. Getting started

4.1. Initial Configuration

The TANDBERG Gatekeeper requires some configuration before it can be used. This must be done using a PC connected to the serial port (Data 1) or by connecting to the system's default IP address: 192.168.0.100.

The IP address, subnet mask and gateway must be configured before use. The Gatekeeper has to be configured with a static IP address. Consult your network administrator for information on which addresses to use.

To set the initial configuration:

1. Connect the supplied null-modem RS-232 cable from Data 1 to a PC running a terminal program.

2. Start a terminal program and configure it to use the serial port with baud rate 115200, 8 data bits, no parity, 1 stop bit, no flow control.

3. Power on the unit if it is not already on.

   You should see the unit display start up information.

   After approximately 2 minutes you will get a login prompt:

   ```
   (none) login: admin
   Password:
   ```

4. Enter the username *admin* and your password. The default password is *TANDBERG*.

   You will be prompted if you want to run the install wizard:

   ```
   Run install wizard [n]: y
   ```

5. Type `y` and press Enter.

6. Specify the following:

   a. The password you want to use for your system. See *Administrator Account* (section 4.2.4) for account details.

   b. The IP address of the system.

   c. The IP subnet mask of the system.

   d. The IP default gateway of the system.

   e. The Ethernet speed.

   f. The local zone prefix, if any, you want to use for the zone controlled by this system. (You should use a local zone prefix if you have a structured dial plan using E.164 aliases. See *Neighboring and dial plans* (section 4.6.1) for more information.

   g. Whether you want to use SSH to administer the system.

   h. Whether you want to use Telnet to administer the system.

7. You will be prompted to log in again. You should see a welcome message like this:

   ```
   Welcome to
   TANDBERG Gatekeeper Release N6.0
   SW Release Date: 2008-03-11
   OK
   ```

8. Login with the username *admin* and your password.

9. Review other system settings. You may want to set the following:

   a. The name of the Gatekeeper. This is used by the TANDBERG Management Suite (TMS) to identify the Gatekeeper. See the `xConfiguration SystemUnit` command (section 17.2.18) for more information on setting the name.

   b. Automatic discovery. If you have multiple Gatekeepers in the same network you may want to disable automatic discovery on some of them. See the `xConfiguration Gatekeeper AutoDiscovery` command (section 17.2.4).

   c. The DNS server address and the domain name (if the Gatekeeper will be configured with hostnames instead of IP address or if URI dialing is required). See the `xConfiguration IP DNS Server Address` command (17.2.6) for more information.

10. To make your new settings take effect, reboot the Gatekeeper by typing the command `xCommand boot`.

11. Disconnect the serial cable.

Note: To securely manage the Gatekeeper you should disable HTTP and Telnet, using the encrypted HTTPS and SSH protocols instead. For increased security, disable HTTPS and SSH as well, using the serial port to manage the system.

Note: If you do not have an IP gateway, configure the Gatekeeper with an unused IP address that is valid in your subnet.

4.2. System Administration

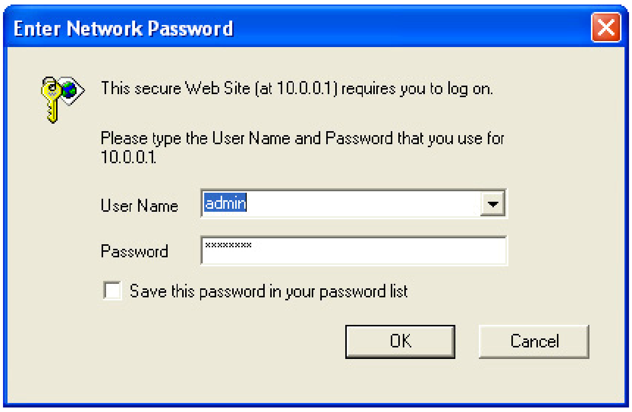
To configure and monitor the TANDBERG Gatekeeper you can either use the web interface or a command line interface.

4.2.1. Web interface

To use the web interface, open a browser window and in the address line type either:

- the IP address of the system
- the system’s host name (if configured in the local DNS server).

You will be presented with the following screen:



Enter Network Password

This secure Web Site (at 10.0.0.1) requires you to log on.

Please type the User Name and Password that you use for 10.0.0.1

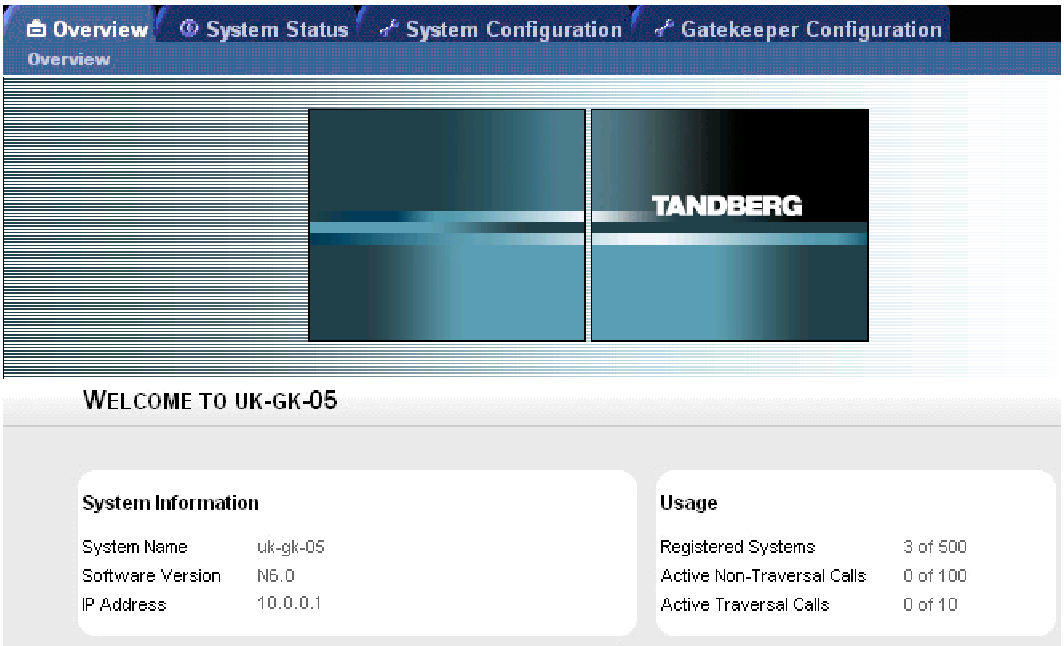
User Name admin

Password ********

□ Save this password in your password list

Enter the User Name *admin* and your system password and select OK.

You will be presented with the Overview screen:



WELCOME TO UK-GK-05

System Information

System Name	uk-gk-05
Software Version	N6.0
IP Address	10.0.0.1

Usage

Registered Systems	3 of 500
Active Non-Traversal Calls	0 of 100
Active Traversal Calls	0 of 10

Note: HTTP and HTTPS must be enabled in order to use the web interface. This is done using the following commands:

```
xConfiguration HTTP Mode: <On/Off>
xconfiguration HTTPS Mode: <On/Off>
```

Note: If web access is required, you are recommended to enable HTTPS and disable HTTP for improved security.

Uploading an HTTPS Server Certificate

For added security, you can upload a PEM file that contains the server certificate used for HTTPS connections to the Gatekeeper from administrator web browsers. You can also upload a PEM file that identifies the private key used to encrypt the server certificate used by the Gatekeeper. This private key must not be password protected.

To upload the HTTPS server certificate files, navigate to *Gatekeeper Configuration* > *Files*. In the Server Certificate section, browse to the appropriate file(s) and then select Upload.

Note: Installation of the HTTPS server certificate files cannot be done via the command line interface.

4.2.2. Command line interface

The command line interface is available over SSH, Telnet and through the serial port.

To use the command line interface, start a session and login with user name *admin* and your password.

- To obtain Help for a particular command, type “?” after the command.
- To complete a word typed into the CLI, or to obtain a list of sub-commands for a particular command, press the TAB key.

The interface groups information in different commands:

`xstatus`

Provides a read only interface to determine the current status of the system. Information such as current calls and registrations is available through this command group.

`xconfiguration`

A read/write interface to set system configuration data such as IP address and subnet.

`xcommand`

A miscellaneous group of commands for setting information or obtaining it.

`xhistory`

Provides historical information about calls and registrations.

`xfeedback`

An event interface, providing information about calls and registrations.

See the *Command Reference* (section 17) for a full list of commands.

Note: SSH and/or Telnet access must be enabled in order to use the command line interface. This is done using the following commands:

```
xConfiguration SSH Mode: <On/Off>
xconfiguration Telnet Mode: <On/Off>
```

Note: For secure operation you should use SSH in preference to Telnet.

4.2.3. Session timeout

By default, administration sessions remain active until you logout. Session timeouts may be enabled using the command:

```
xConfiguration Session TimeOut
```

or using the web interface via *System Configuration* > *System* and in the Services section entering a value in the Session time out (minutes) field.

4.2.4. Administrator Account

All administration requires you to log in to the administration account with the user name *admin* and a password. The default password is *TANDBERG*, which you are recommended to change as soon as possible. Choose a strong password, particularly if administration over IP is enabled.

The password can be changed on the web interface via *System Configuration* > *System* or through the command line interface using the command:

```
xconfiguration systemunit password: new_password
```

If you forget your password, it is possible to set a new password using the following procedure:

1. Reboot the Gatekeeper.
2. Connect to the Gatekeeper over the serial interface once it has restarted.
3. Login with the user name *pwrec*. No password is required.
4. You will be prompted for a new password.

Note: The pwrec account is only active for one minute following a restart. Beyond that time you will have to restart the system again to change the password. Because access to the serial port allows the password to be reset, it is recommended that you install the Gatekeeper in a physically secure environment.

4.2.5. Root Account

The Gatekeeper provides a root account with the same password as the admin account. This 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.

4.3. Backups

You are recommended to maintain a backup of your Gatekeeper configuration. Using the command line interface, log on to the Gatekeeper as *admin* and type `xConfiguration`. Save the resulting output to a file, using cut-and-paste or some other means provided by your terminal emulator. Pasting this information back in to the command line shell will restore your configuration.

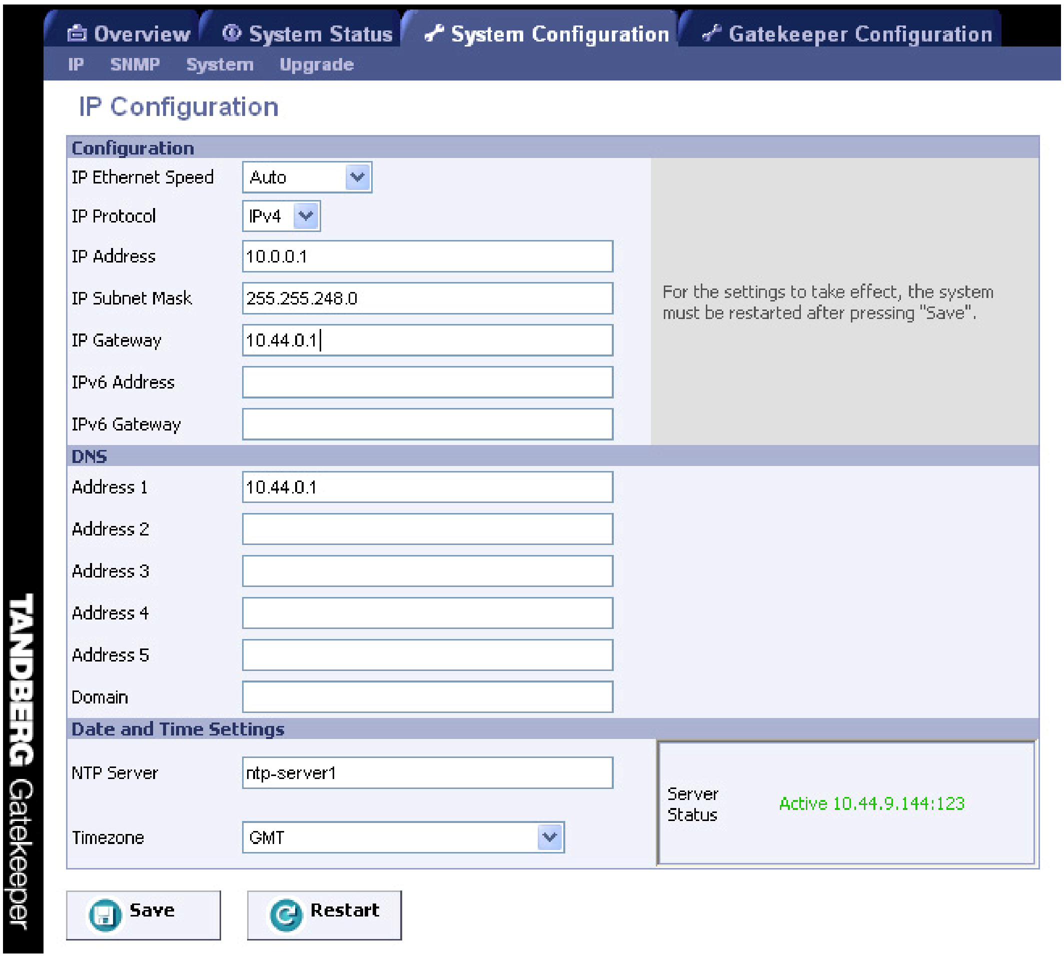
4.4. IP Configuration

The Gatekeeper may be configured to use IPv4, IPv6 or both protocols. If using both protocols, the Gatekeeper will act as a gateway if necessary, allowing calls to be made between an IPv4-only endpoint and an IPv6-only endpoint. This behavior will use a traversal license for each call gatewayed between IPv4 and IPv6.

IPv4 and IPv6 dual stack behavior is controlled by the command:

```
xConfiguration IPProtocol: <Both/IPv4/IPv6>
```

or using the web interface via *System Configuration* > *IP Configuration* shown in Figure 3 below:



IP Configuration

Configuration

IP Ethernet Speed	Auto
IP Protocol	IPv4
IP Address	10.0.0.1
IP Subnet Mask	255.255.248.0
IP Gateway	10.44.0.1
IPv6 Address	
IPv6 Gateway	

For the settings to take effect, the system must be restarted after pressing "Save".

DNS

Address 1	10.44.0.1
Address 2	
Address 3	
Address 4	
Address 5	
Domain	

Date and Time Settings

NTP Server	ntp-server1
Timezone	GMT

Server Status: Active 10.44.9.144:123

Figure 3: Selecting IP Protocol

4.5. Endpoint Registration

Before an endpoint can use the Gatekeeper it must first register with it.

There are two ways an endpoint can register:

- Automatically
- Manually by specifying the IP address of the Gatekeeper.

Note: You can disable automatic registration on the Gatekeeper. See the *Auto Discovery* command (section 17.2.4) for more information.

When registering, the endpoint registers with one or more of the following:

- One or more H.323 IDs
- One or more E.164 aliases.

Users of other registered endpoints can then call the endpoint by using either the H.323 ID, a URI, an E.164 alias, or one of the services.

By default, if you attempt to register an alias which has already been registered with the system, your registration will be rejected. This helps you to identify when two users have a conflicting alias.

In some deployments an endpoint may frequently receive a new IP address, causing unwanted registration rejections. When it tries to register, it may be rejected because the Gatekeeper still has a registration from its old IP address. The Gatekeeper may be configured to allow an endpoint to overwrite the old IP address. To do this, either issue the command:

```
xConfiguration Gatekeeper Registration ConflictMode: <Overwrite/Reject>
```

or go to *Gatekeeper Configuration* > *Restrictions* and in the Policy section, from the Registration conflict policy drop-down menu select Overwrite.

Consult the endpoint documentation for information on how to configure it with a Gatekeeper.

Note: When URI dialing is used to discover an endpoint, the URI used is based on either the H.323 ID or the E.164 alias that the endpoint registered with. The local domain is then added to this. For more information see *URI Dialing* (section 9).

4.6. Neighbor Gatekeepers

4.6.1. Neighboring and dial plans

As you start deploying more than one Gatekeeper or Border Controller, it is useful to neighbor the systems together so that they can exchange information about registered endpoints. Each Gatekeeper or Border Controller forms an H.323 zone and is responsible for the endpoints within that zone. There are a number of ways this can be done, depending on the complexity of your system.

Flat dial plan

The simplest approach is to assign each endpoint a unique alias and divide the endpoint registrations between the Gatekeepers and Border Controllers. Each Gatekeeper or Border Controller is then configured with the addresses of all other Gatekeepers and Border Controllers. When a system receives a call for an endpoint which is not registered with it, it will send out a Location Request to all the other Gatekeepers and Border Controllers on the system. Whilst conceptually simple, this sort of flat dial plan does not scale very well: adding or moving a Gatekeeper requires changing the configuration of every Gatekeeper and Border Controller; one call attempt can result in a large number of location requests.

Structured dial plan

An alternative deployment would use a structured dial plan whereby endpoints are assigned an alias based on the system they are registering with. Using E.164 aliases, each Gatekeeper or Border Controller would be assigned an area code. When the Gatekeepers and Border Controllers are neighbored together, each neighbor is configured with its corresponding area code as a prefix. That neighbor will now 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 neighbors 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 Gatekeeper should be configured to strip prefixes before placing the Location Request.

A structured dial plan will minimize the number of location requests issued when a call is attempted, but, as described above, still requires a fully connected mesh of all Gatekeepers and Border Controllers in your deployment. A hierarchical dial plan (see below) can simplify this.

Hierarchical dial plan

One Gatekeeper is nominated as the directory gatekeeper for the deployment. All Border Controllers and public Gatekeepers are neighbored with it and vice versa. There is no need to neighbor the Border Controllers and public Gatekeepers with each other. Adding a new Border Controller or public Gatekeeper now only requires changing configuration on that system and the Directory Gatekeeper.

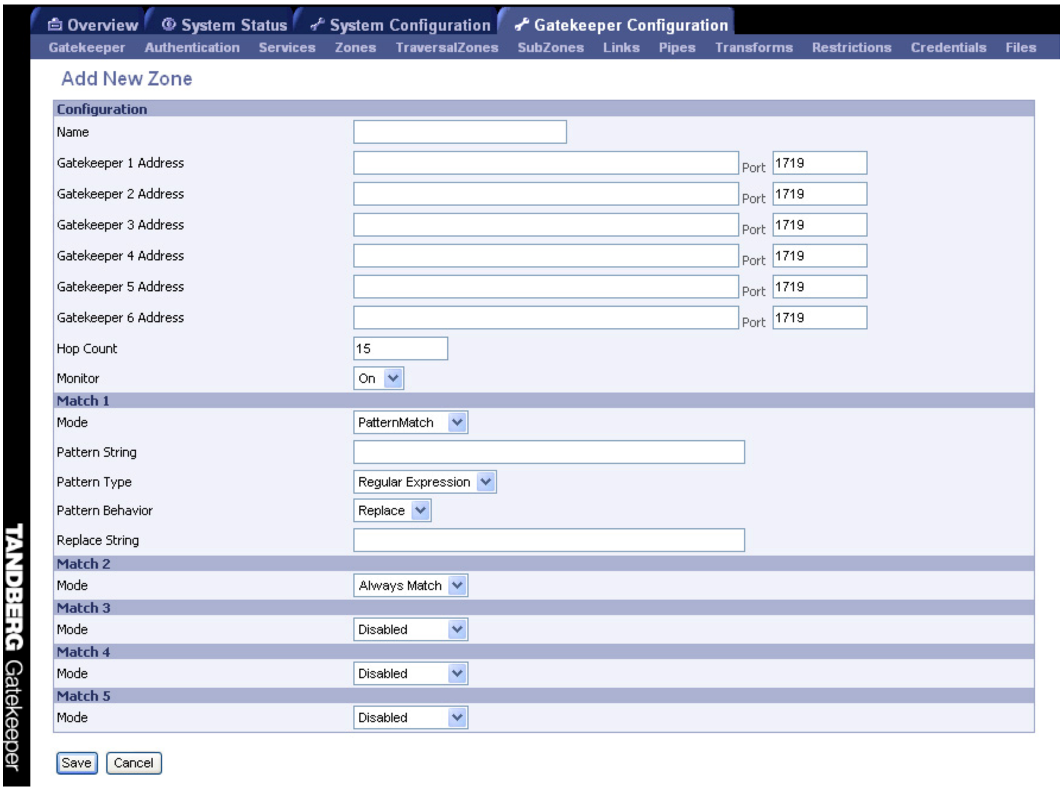
Failure of the directory gatekeeper could cause significant disruption to communications. Consideration should be given to the use of Alternate Gatekeepers (see section 4.7) for increased resilience.

4.6.2. Adding Neighbors and configuring zones

Neighbors are added and zones configured through the command line interface using the `xconfiguration zones` family of commands and `xCommand ZoneAdd` or through the web interface via *Gatekeeper Configuration* > *Zones* - either select Add New Zone, or highlight an existing zone and select Edit, to access the screen shown in Figure 4.

The prefixes and suffixes described above are formed using patterns: each zone may have up to 5 patterns assigned, each of which may be defined as a prefix or a suffix.

Patterns are not used, and not displayed on the web interface, if the pattern match mode is set to `always` or `disabled`.



Add New Zone

Configuration			
Name			
Gatekeeper 1 Address		Port	1719
Gatekeeper 2 Address		Port	1719
Gatekeeper 3 Address		Port	1719
Gatekeeper 4 Address		Port	1719
Gatekeeper 5 Address		Port	1719
Gatekeeper 6 Address		Port	1719
Hop Count	15		
Monitor	On		
Match 1			
Mode	PatternMatch		
Pattern String			
Pattern Type	Regular Expression		
Pattern Behavior	Replace		
Replace String			
Match 2			
Mode	Always Match		
Match 3			
Mode	Disabled		
Match 4			
Mode	Disabled		
Match 5			
Mode	Disabled		

Figure 4: Adding a new zone

4.6.3. Search Order

If a called alias matches a prefix or suffix zone a strong match is achieved. A weak match is achieved if a zone is to be queried only because it has no pattern matching configured.

When an incoming call request is received a Gatekeeper will first search all of its registered endpoints. If no match is found, all strongly matching neighbor and traversal zones will be queried concurrently. If the target is not found in any of the strongly matching zones, all weakly matching neighbor zones will be queried, then all weakly matching traversal zones. Finally, if a match has still not been found, a DNS query may be attempted (see section 9).

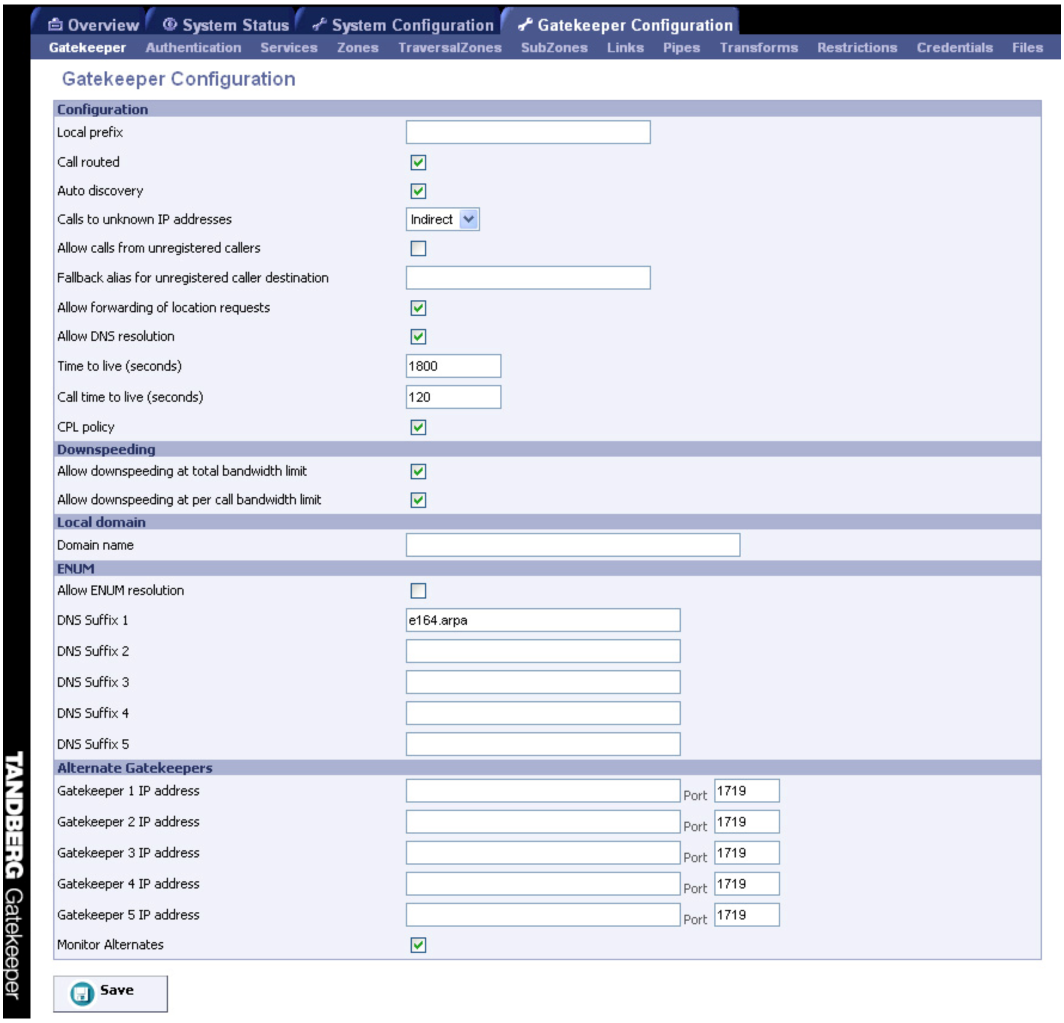
4.7. Alternates

Alternate Gatekeeper support is provided to increase the reliability of your deployment. If one Gatekeeper becomes unavailable, perhaps due to a network or power outage, another will be used as an Alternate. Alternates share responsibility for their endpoint community: an individual endpoint may be registered with any one of the Alternates. You should configure Alternates identically for all registration and call features such as authentication, bandwidth control and policy. If you do not do this, endpoint behavior will vary unpredictably depending on which Alternate it is currently registered with. Alternates should also be deployed on the same LAN as each other so that they may be configured with the same routing information such as local domain names and local domain subnet masks.

Each Gatekeeper may be configured with the IP addresses of up to five Alternates. When an endpoint registers with the Gatekeeper, it is presented with the IP addresses of all the Alternates. If the endpoint loses contact with its initial Gatekeeper, it will seek to register with one of the Alternates. This may result in your endpoint community's registrations being spread over all the Alternates.

When a Gatekeeper receives a Location Request, if it cannot respond from its own registration database, it will query all of its Alternates before responding. This allows the pool of registrations to be treated as if they were registered with a single Gatekeeper.

The Alternate Gatekeepers can be configured within the web interface via *Gatekeeper Configuration* > *Gatekeeper* within the Alternate Gatekeepers section (see Figure 5).



Gatekeeper Configuration

Configuration			
Local prefix			
Call routed	☑		
Auto discovery	☑		
Calls to unknown IP addresses	Indirect		
Allow calls from unregistered callers	□		
Fallback alias for unregistered caller destination			
Allow forwarding of location requests	☑		
Allow DNS resolution	☑		
Time to live (seconds)	1800		
Call time to live (seconds)	120		
CPL policy	☑		
Downspeeding			
Allow downspeeding at total bandwidth limit	☑		
Allow downspeeding at per call bandwidth limit	☑		
Local domain			
Domain name			
ENUM			
Allow ENUM resolution	□		
DNS Suffix 1	e164.arpa		
DNS Suffix 2			
DNS Suffix 3			
DNS Suffix 4			
DNS Suffix 5			
Alternate Gatekeepers			
Gatekeeper 1 IP address		Port	1719
Gatekeeper 2 IP address		Port	1719
Gatekeeper 3 IP address		Port	1719
Gatekeeper 4 IP address		Port	1719
Gatekeeper 5 IP address		Port	1719
Monitor Alternates	☑		

Figure 5: Alternate Gatekeeper configuration

4.8. Call Processing Overview

Figure 6 illustrates the process the Gatekeeper performs when receiving call requests.

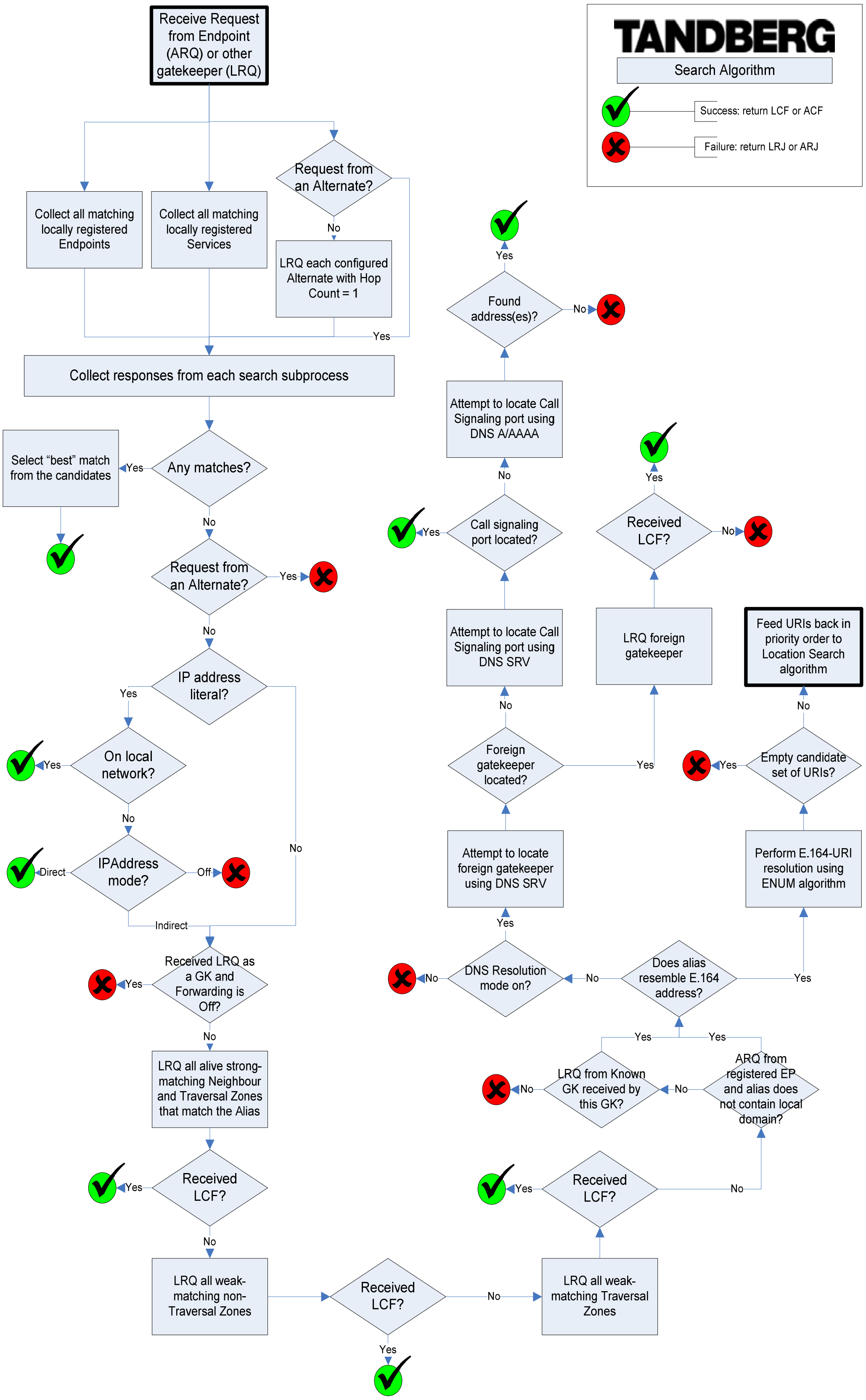

TANDBERG
Search Algorithm
Success: return LCF or ACF
Failure: return LRJ or ARJ
Receive Request from Endpoint (ARQ) or other gatekeeper (LRQ)
Collect all matching locally registered Endpoints
Collect all matching locally registered Services
Request from an Alternate?
No
LRQ each configured Alternate with Hop Count = 1
Yes
Collect responses from each search subprocess
Any matches?
Yes
Select "best" match from the candidates
No
Request from an Alternate?
Yes
No
IP address literal?
Yes
On local network?
Yes
No
IPAddress mode?
Direct
Off
Indirect
No
Received LRQ as a GK and Forwarding is Off?
Yes
No
LRQ all alive strong-matching Neighbour and Traversal Zones that match the Alias
Received LCF?
Yes
No
LRQ all weak-matching non-Traversal Zones
Received LCF?
Yes
No
LRQ all weak-matching Traversal Zones
Received LCF?
Yes
No
ARQ from registered EP and alias does not contain local domain?
Yes
No
LRQ from Known GK received by this GK?
No
Yes
Does alias resemble E.164 address?
No
Yes
DNS Resolution mode on?
No
Yes
Attempt to locate foreign gatekeeper using DNS SRV
Foreign gatekeeper located?
Yes
No
Attempt to locate Call Signaling port using DNS SRV
Call signaling port located?
Yes
No
Attempt to locate Call Signaling port using DNS A/AAAA
Found address(es)?
Yes
No
LRQ foreign gatekeeper
Received LCF?
Yes
No
Perform E.164-URI resolution using ENUM algorithm
Empty candidate set of URIs?
Yes
No
Feed URIs back in priority order to Location Search algorithm

Figure 6: Location decision flow diagram

When an endpoint wants to call another endpoint it presents the address it wants to call to the Gatekeeper using a protocol knows as RAS. The Gatekeeper applies any transforms (see section 5), tries to resolve the address, and if successful supplies the calling endpoint with information about the called endpoint.

The destination address can take several forms: IP address, H.323 ID, E.164 alias or a full H.323 URI.

When an H.323 ID or E.164 alias is used, the Gatekeeper looks for a match between the dialed address and the aliases registered by its endpoints. If no match is found, it may query other Gatekeepers and Border Controllers.

When dialing by H.323 URI, the destination address resembles an email address. The Gatekeeper first follows the procedure for matching H.323 IDs. If that fails it looks for a Gatekeeper or Border Controller responsible for the domain (the part of the URI following the `@` symbol) and queries that device.

Dialing by IP address is necessary when the destination endpoint is not registered with a Gatekeeper or Border Controller. If it is registered, then one of the other addressing schemes should be used instead as they are more flexible. From your registered endpoint, dial the IP address of the endpoint you wish to call. This requires that the Gatekeeper has `xConfiguration Gatekeeper CallsToUnknownIPAddresses` correctly configured (see section 17.2.4).

When one endpoint calls another, the Gatekeeper is involved in locating the called endpoint. By default, once the endpoint is located, the Gatekeeper takes no further part in the call signaling. By enabling call routed mode, all call signaling will be routed through the Gatekeeper. This is useful if you need accurate information about call start and stop times. Call Detail Records (CDRs) may be extracted from the Gatekeeper event log.

Note: Traversal calls are always call routed, regardless of the setting of Call Routed Mode.

5. Transforming Destination Aliases

5.1. Alias Transforms

The Alias Transforms function takes any aliases present in ARQ and LRQ messages and runs a set of transformations on them. The resulting aliases will then be used in the normal Gatekeeper logic, exactly as if those aliases were unchanged. Alias transforms will be applied prior to possible CPL modification and Zone transforms. The Alias transforms will not have any effect on aliases presented in GRQ or RRQ messages.

Alias transform rules are created either:

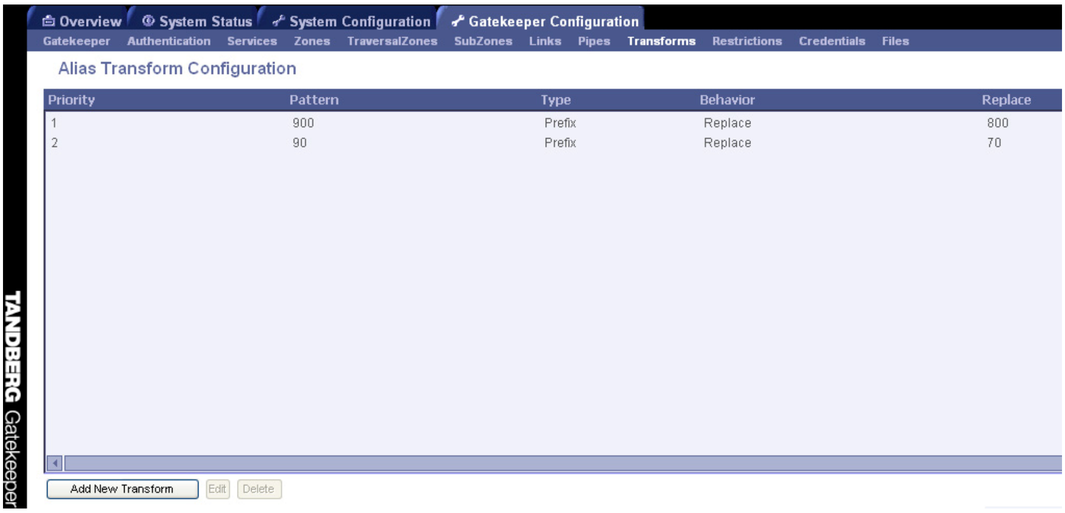
- using the `xconfiguration Gatekeeper Transform` commands, or
- using the web interface via *Gatekeeper Configuration* > *Transforms* and selecting Add New Transform.

Alias transforms support the use of Regular Expressions. See *Appendix C* for further information.

Example

We have two gateways registered with the Gatekeeper with prefixes of 7 and 8 respectively.

We want to allow the users to dial 9 for an “outside line”, but use GW1 for local calls, and GW2 for international calls. We should allow an alias manipulation that takes a destination alias of 90047... and replaces it with 80047... and an alias of 90118... with 70118.... This is achieved by configuring alias transforms as shown in Figure 7:



Alias Transform Configuration

Priority	Pattern	Type	Behavior	Replace
1	900	Prefix	Replace	800
2	90	Prefix	Replace	70

Figure 7: Example configuration of alias transforms

5.2. Zone Transforms

It is possible to direct an incoming location request to a different alias by replacing either the prefix or the suffix of the alias with a new string, or by using regular expressions to specify the way in which the alias is to be transformed.

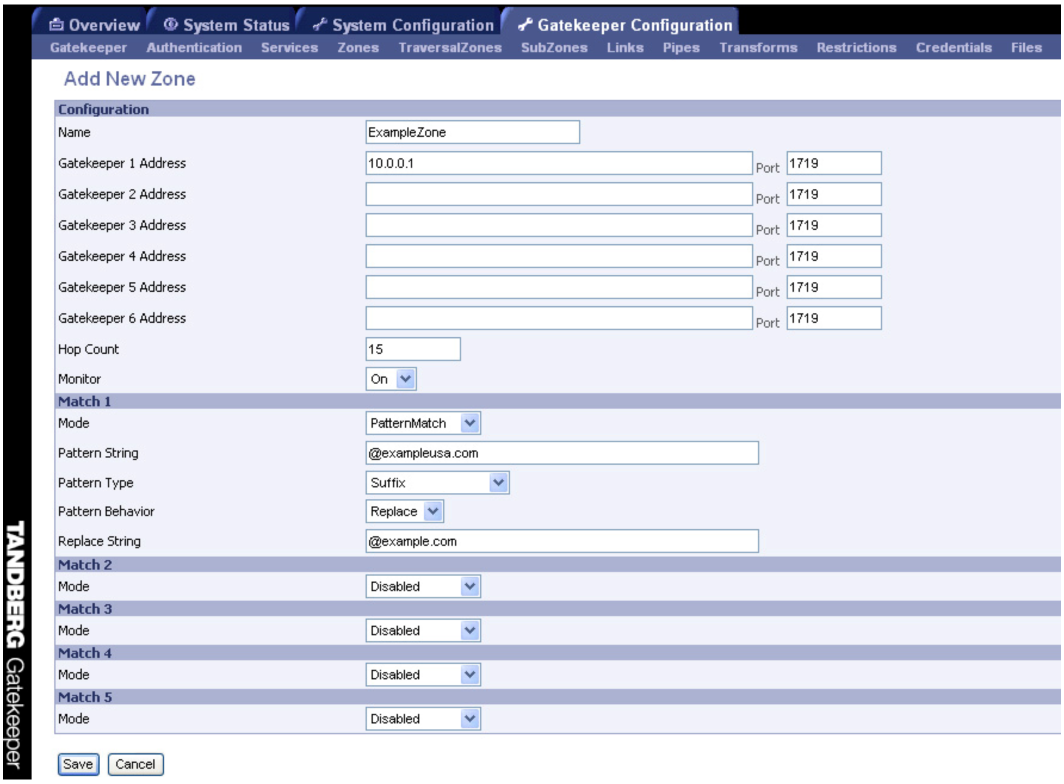
Zone transform rules are created either:

- using the `xconfiguration zones` set of commands, or
- using the web interface when adding or editing a zone via *Gatekeeper Configuration* > *Zones*. You must first select from the Match 1, Match 2, etc. sections a Mode of PatternMatch in order to access the options (see Figure 4).

Zone transforms support the use of Regular Expressions. See *Appendix C* for more information.

Example

Endpoints might be registered to a Gatekeeper with aliases of the form user@example.com. If someone were to dial user@exampleusa.com we might want to try and find that user as user@example.com, hence we need a rule that replaces the suffix exampleusa.com with example.com before searching off the box. This can be achieved by configuring the zone transforms as shown in Figure 8:



Add New Zone

Configuration			
Name	ExampleZone		
Gatekeeper 1 Address	10.0.0.1	Port	1719
Gatekeeper 2 Address		Port	1719
Gatekeeper 3 Address		Port	1719
Gatekeeper 4 Address		Port	1719
Gatekeeper 5 Address		Port	1719
Gatekeeper 6 Address		Port	1719
Hop Count	15		
Monitor	On		
Match 1			
Mode	PatternMatch		
Pattern String	@exampleusa.com		
Pattern Type	Suffix		
Pattern Behavior	Replace		
Replace String	@example.com		
Match 2			
Mode	Disabled		
Match 3			
Mode	Disabled		
Match 4			
Mode	Disabled		
Match 5			
Mode	Disabled		

Figure 8: Example configuration of zone transforms

6. Unregistered Endpoints

Although most calls are made between endpoints registered with a Gatekeeper or Border Controller, it is sometimes necessary to place a call to or from an unregistered endpoint.

6.1. Calling from an Unregistered Endpoint

An unregistered endpoint can call an endpoint registered with the Gatekeeper. If there are no firewalls between the unregistered endpoint and the called endpoint, it is possible (though not recommended) to place the call by dialing the target endpoint's IP address. A better way of placing the call from an unregistered endpoint is to pass the alias of the called endpoint to the Gatekeeper. The Gatekeeper will then resolve the alias and place the call as normal.

Not all endpoints allow you to enter an alias and an IP address to which the call should be placed. In that case you can simply place the call to the IP address of the Gatekeeper, with no alias information. The Gatekeeper may be configured to associate all such anonymous calls with a single destination alias. This is achieved with the command:

```
xConfiguration Gatekeeper Unregistered Caller Fallback: <destination>
```

or using the web interface via *Gatekeeper Configuration* > *Gatekeeper Configuration* and entering the alias in the Fallback alias for unregistered caller destination field.

6.2. Calling to an Unregistered Endpoint

Calls can be placed to an unregistered endpoint by dialing its IP address or (if the DNS system has been appropriately configured) using an H.323 URI.

If URI dialing is used, DNS is queried for a call signaling address and, if found, the call is placed to that address. See *URI Dialing* (section 9) for details of how to configure the Call Signaling SRV Record.

It is sometimes undesirable for a system to place a call to an IP address directly. Instead, you may want a neighbor to place the call on behalf of the Gatekeeper. You can configure this on the Gatekeeper using the command:

```
xConfiguration Gatekeeper CallsToUnknownIPAddresses:
<Off/Indirect/Direct>
```

or using the web interface via *Gatekeeper Configuration* > *Gatekeeper* and from within the Configuration section selecting the desired option from the Calls to unknown IP addresses drop-down menu.

There are three possible settings:

`Direct`

This setting will allow the endpoint to make the call to the unknown IP address without 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 the call the Gatekeeper will check to see if the address belongs to one of its local subzones. If so, it will allow the call. If not, it will query its neighbors for the remote address, relying on the response from the neighbor to allow the ability for the call to be completed; connecting through the routing rules as it would through the neighbor relationship.

`Off`

This will not allow any endpoint registered directly to the Gatekeeper to call an IP address of any system not also registered directly to that Gatekeeper.

The default is `Indirect`.

When the Gatekeeper is used with a Border Controller for firewall traversal, you will typically set `CallsToUnknownIPAddresses` to `Indirect` on the Gatekeeper and `Direct` on the Border Controller. This will allow endpoints registered to the gatekeeper to successfully traverse the firewall in order to call public endpoints on the Internet. This is described in more detail in *Dialing Public IP Addresses* (section 11.3).

7. Bandwidth Control

7.1. About Bandwidth Control

The TANDBERG Gatekeeper allows you to control endpoints' use of bandwidth on your network.

Figure 9 shows a typical network deployment: a broadband LAN, where high bandwidth calls are acceptable; a pipe to the internet with restricted bandwidth; and two satellite offices, each with their own restricted pipes.

In order to utilize the available bandwidth efficiently, the TANDBERG Gatekeeper allows you to model your network, and bandwidth controls on individual components of the network. Bandwidth controls may be set on a call-by-call basis and on a total concurrent usage basis.

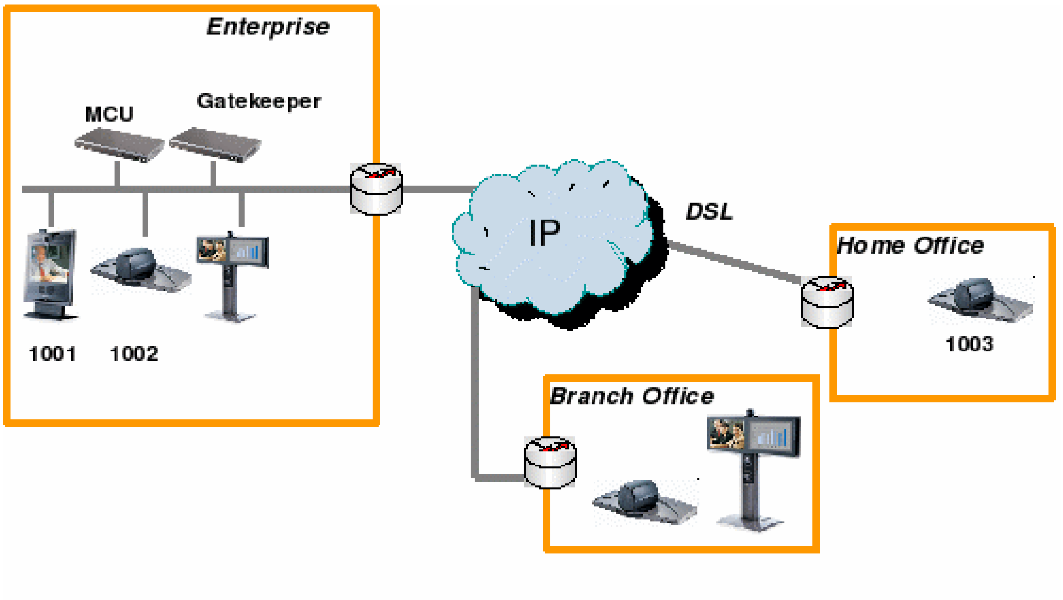


Figure 9: Typical network deployment

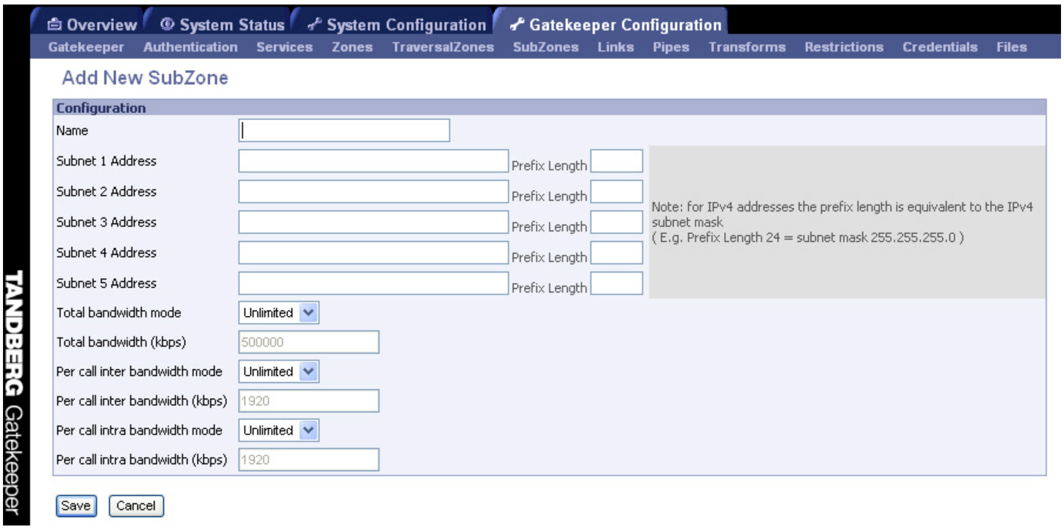
7.2. Subzones

All endpoints registered with your Gatekeeper are part of its local zone. As shown in Figure 9, the local zone can contain two or more different networks with different bandwidth limitations. In order to model this, the local zone is made up of one or more subzones. When an endpoint registers with the Gatekeeper it is assigned to a subzone, based on its IP address.

By default all endpoints registering with the Gatekeeper are assigned to the default subzone. This is suitable if you have uniform bandwidth available between all your endpoints. When you have differing bandwidth provision, as in Figure 9, you should create a new subzone for each pool of endpoints. Each subzone you create can include up to 5 subnets (based on a specified range of IP addresses).

Subzones are added and configured using the web interface via *Gatekeeper Configuration* >. *SubZones*, and the either selecting Add New SubZone, or highlighting an existing subzone and selecting Edit. This will take you to the screen shown in Figure 10. You can also add and configure subzones using the following commands:

```
xConfiguration SubZones SubZone [1..100] Name
xConfiguration SubZones SubZone [1..100] Subnet [1..5] IP Address
xConfiguration SubZones SubZone [1..100] Subnet [1..5] IP Prefixlength
```



Add New SubZone

Configuration

Name			
Subnet 1 Address		Prefix Length	
Subnet 2 Address		Prefix Length	
Subnet 3 Address		Prefix Length	
Subnet 4 Address		Prefix Length	
Subnet 5 Address		Prefix Length	
Total bandwidth mode	Unlimited		
Total bandwidth (kbps)	500000		
Per call inter bandwidth mode	Unlimited		
Per call inter bandwidth (kbps)	1920		
Per call intra bandwidth mode	Unlimited		
Per call intra bandwidth (kbps)	1920		

Note: for IPv4 addresses the prefix length is equivalent to the IPv4 subnet mask
(E.g. Prefix Length 24 = subnet mask 255.255.255.0)

Figure 10: Configuring a SubZone

7.2.1. Subzone Bandwidths

Each subzone may be configured with its own bandwidth limits. Calls placed between two endpoints in the same subzone consume resource from the subzone's allocation. Subzone bandwidths are configured on the *Gatekeeper Configuration* > *SubZones* page (see Figure 10 for a screenshot of the configuration) or using the following command line commands:

```
xConfiguration SubZones SubZone [1..100] Bandwidth Total Mode
xConfiguration SubZones SubZone [1..100] Bandwidth Total Limit
xConfiguration SubZones SubZone [1..100] Bandwidth PerCall Mode
xConfiguration SubZones SubZone [1..100] Bandwidth PerCall Limit
```

7.2.2. Subzone links

Subzones may be configured with links joining them to each other and to other zones. These links are used to calculate how a call is routed over the network and so which zones and subzones are involved. If multiple routes are possible, your Gatekeeper will select the one with the fewest links.

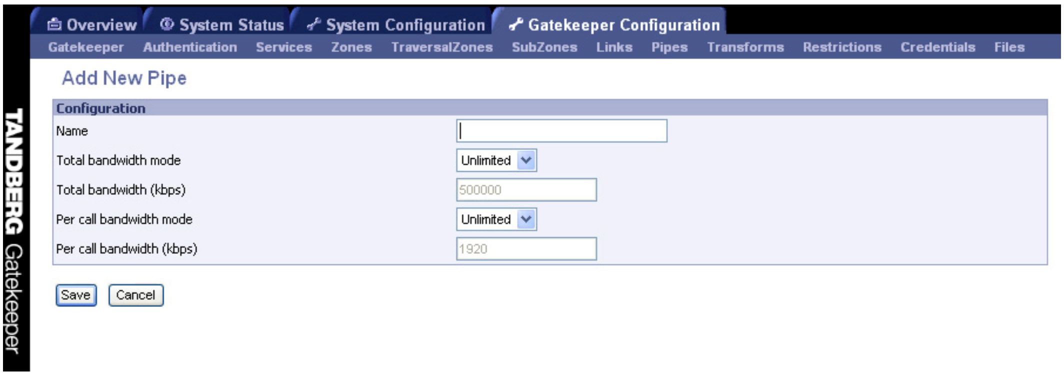
Links may be configured using the web interface via *Gatekeeper Configuration* > *Links*, or via the command line using the following commands:

```
xConfiguration Links Link [1..100] Name
xConfiguration Links Link [1..100] Node1 Name
xConfiguration Links Link [1..100] Node2 Name
xConfiguration Links Link [1..100] Pipe1 Name
xConfiguration Links Link [1..100] Pipe2 Name
```

7.2.3. Pipes

When calls are placed between endpoints in different subzones, it is possible to control the bandwidth used on the link between them. To do this, create a pipe and configure it with the required bandwidth characteristics. This pipe is then assigned to a link. Calls traversing the link will now take the pipe's bandwidth allocation into consideration. Pipes are created and configured via *Gatekeeper Configuration* > *Pipes* and then either selecting Add New Pipe, or highlighting an existing pipe and selecting Edit. The screen shown in Figure 11 will then appear. You can also create and configure pipes using the following commands:

```
xConfiguration Pipes Pipe [1..100] Name
xConfiguration Pipes Pipe [1..100] Bandwidth Total Mode
xConfiguration Pipes Pipe [1..100] Bandwidth Total Limit
xConfiguration Pipes Pipe [1..100] Bandwidth PerCall Mode
xConfiguration Pipes Pipe [1..100] Bandwidth PerCall Limit
```



Add New Pipe

Configuration	
Name	
Total bandwidth mode	Unlimited
Total bandwidth (kbps)	500000
Per call bandwidth mode	Unlimited
Per call bandwidth (kbps)	1920

Figure 11: Configuring a pipe

Pipes may be shared between one or more links. This is used to model the situation where a site communicates with several other sites over the same broadband connection to the Internet.

Each link may have up to two pipes associated with it. This is used to model the situation where two sites each have their own broadband connection to the Internet. In this case, a link between the two sites must go through both broadband connections, and this is modeled using two pipes, one for each connection.

Calls between zones or subzones consume bandwidth from each zone and any pipes on the link between them.

When a Gatekeeper is neighbored with another Gatekeeper or Border Controller, the neighbor is placed in its own zone. This allows you to control the bandwidth used by calls to and from endpoints controlled by the other Gatekeeper. Sometimes you may place and receive calls to Gatekeepers you are not neighbored with (see *URI Dialing*, section 9). These Gatekeepers, and any unregistered endpoints reached by dialing their IP address, are placed in the Default Zone.

7.3. Insufficient Bandwidth

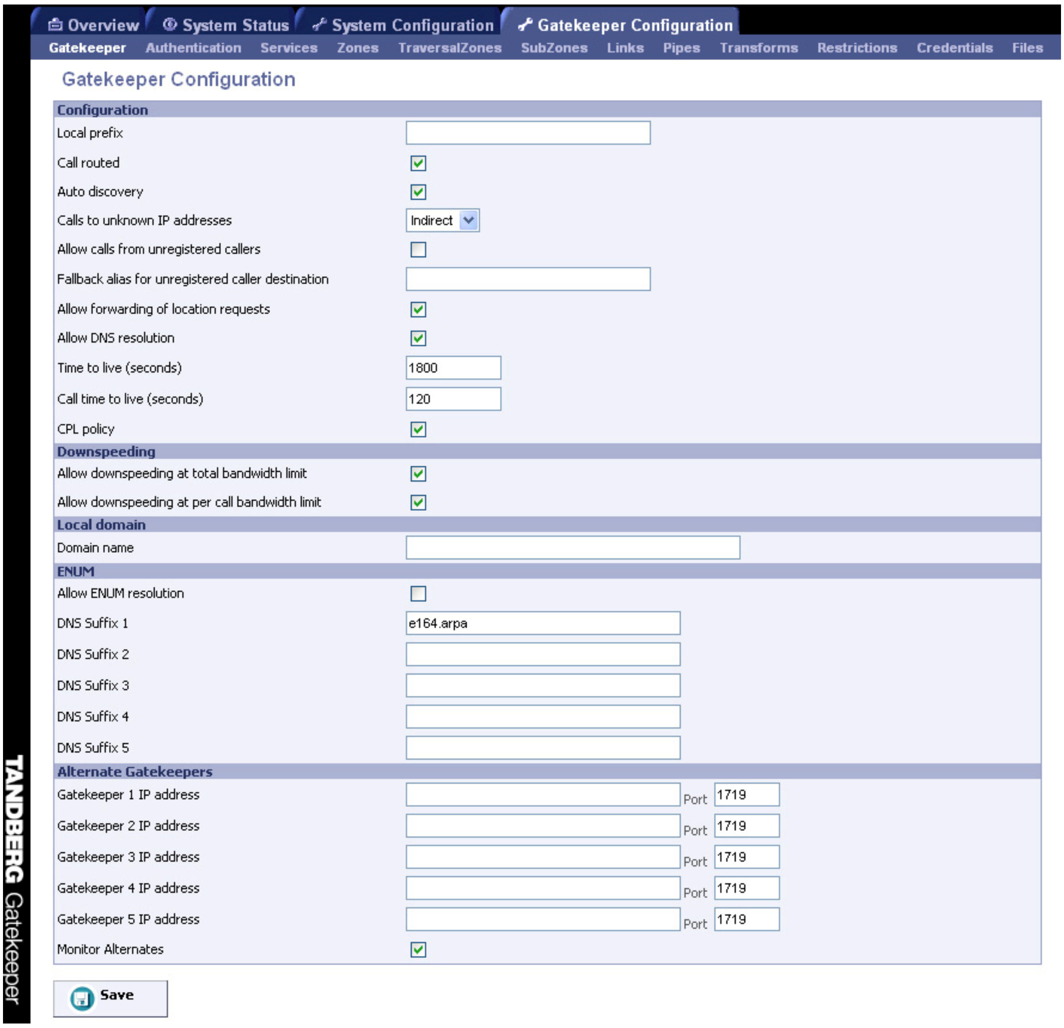
7.3.1. Insufficient bandwidth

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 Gatekeeper will still attempt to connect the call, but at a reduced bandwidth - known as downspeeding .

You can prevent the downspeeding of calls by navigating to *Gatekeeper Configuration* > *Gatekeeper* and in the Downspeeding section, clearing the relevant boxes (see Figure 12). You can also control whether or not calls are downspeeded through the following commands:

```
xConfiguration Gatekeeper Downspeed PerCall Mode: <On/Off>
xConfiguration Gatekeeper Downspeed Total Mode: <On/Off>
```

If downspeeding has been disallowed and there is insufficient bandwidth to place the call at the originally requested rate, the call will not be placed.



Gatekeeper Configuration

Configuration	
Local prefix	
Call routed	☑
Auto discovery	☑
Calls to unknown IP addresses	Indirect
Allow calls from unregistered callers	□
Fallback alias for unregistered caller destination	
Allow forwarding of location requests	☑
Allow DNS resolution	☑
Time to live (seconds)	1800
Call time to live (seconds)	120
CPL policy	☑
Downspeeding	
Allow downspeeding at total bandwidth limit	☑
Allow downspeeding at per call bandwidth limit	☑
Local domain	
Domain name	
ENUM	
Allow ENUM resolution	□
DNS Suffix 1	e164.arpa
DNS Suffix 2	
DNS Suffix 3	
DNS Suffix 4	
DNS Suffix 5	
Alternate Gatekeepers	
Gatekeeper 1 IP address	Port 1719
Gatekeeper 2 IP address	Port 1719
Gatekeeper 3 IP address	Port 1719
Gatekeeper 4 IP address	Port 1719
Gatekeeper 5 IP address	Port 1719
Monitor Alternates	☑

Figure 12: Configuring downspeeding options

7.4. Bandwidth Control and Firewall Traversal

When a Gatekeeper and Border Controller or VCS Expressway are being used to traverse a firewall, an additional subzone and zone(s) come into use on the Gatekeeper, as follows:

- A **traversal zone** is used to represent each zone containing a Border Controller or VCS Expressway used by this Gatekeeper for firewall traversal. A gatekeeper can have up to 50 traversal zones.

- **The traversal subzone** represents the Gatekeeper itself. The traversal subzone allows you to control total and per-call bandwidths passing through the Gatekeeper. Unlike other subzones, no endpoints can be registered in this subzone.

7.5. Bandwidth Control Examples

7.5.1. Example without a firewall

One possible configuration for the deployment in Figure 9 is shown in Figure 13. Each of the offices is represented as a separate subzone, 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.

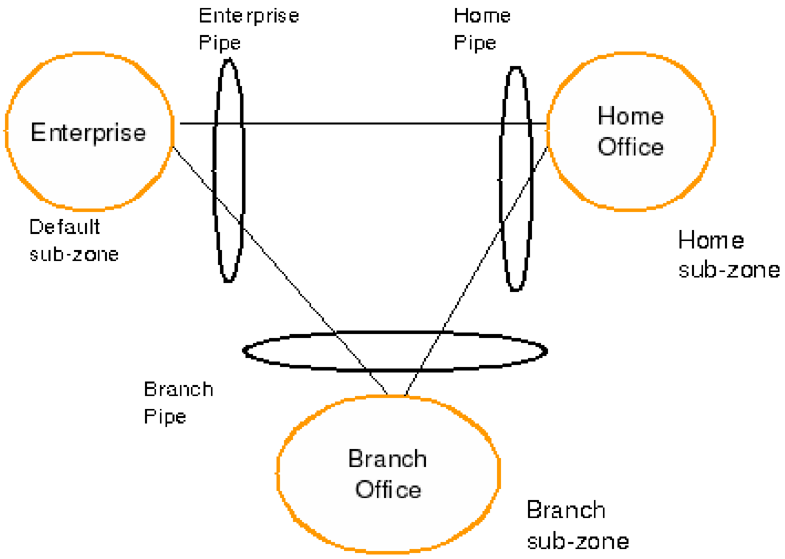


Figure 13: Bandwidth control example

There are no firewalls involved in the scenario shown in Figure 9, so we can configure 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. A call placed between the Home Office and Branch Office will consume bandwidth in the Home and Branch subzones and on the Home and Branch pipe. The enterprise's bandwidth budget will be unaffected by the call.

7.5.2. Example with a firewall

If we modify our deployment to include firewalls between the offices, we can use the firewall traversal capability of the TANDBERG Gatekeeper and Border Controller to maintain connectivity.

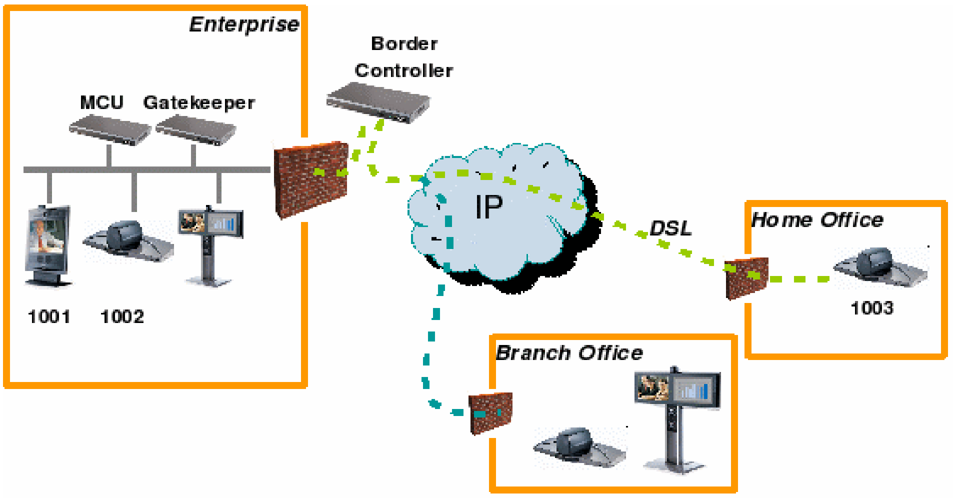


Figure 14: Network deployment with firewalls

In Figure 14, the endpoints in the enterprise register with the Gatekeeper, whilst those in the branch and home office register with the Border Controller.

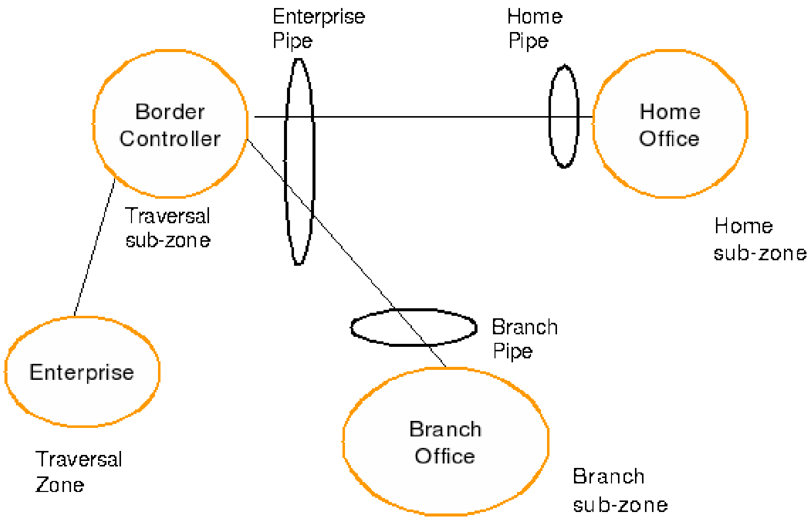


Figure 15: Border Controller example configuration

Figure 15 shows how the Border Controller could be configured for the deployment in Figure 14. 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 Border Controller. This is shown by the absence of a link between the Home and Branch subzones.

The Traversal Zone in Figure 15 represents the Enterprise Gatekeeper. The Border Controller will consume bandwidth from the Traversal Zone for all calls placed to endpoints managed by the Enterprise Gatekeeper. In this example we have assumed that there is no bottleneck on the link between the Border Controller and the Enterprise 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.

The traversal subzone in Figure 15 may be used to control the amount of traffic flowing through the Border Controller itself.

Because the Gatekeeper is only managing endpoints on the LAN, its configuration is simpler as shown in Figure 16.

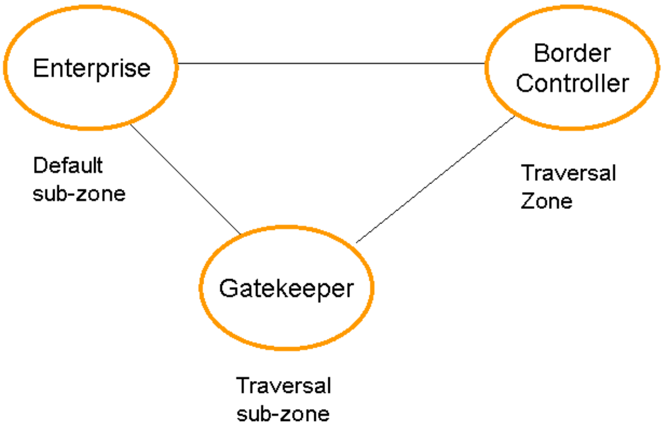


Figure 16: Gatekeeper example configuration

All of the endpoints in the enterprise will be assigned to the default subzone. The Traversal subzone controls traversal traffic flowing through the Gatekeeper, whilst the Traversal Zone controls all traffic traversing the enterprise firewall and passing on to the Border Controller. Both subzones and the Traversal zone are linked: the link between the default subzone and the Traversal zone is used by endpoints which can send media directly to the Border Controller. The other two links are used by endpoints using the Gatekeeper to traverse the firewall.

The Gatekeeper is shipped with Default Zone and Default and Traversal subzones already configured. They are also preconfigured with the links between these zones to allow calls to be placed. You may delete or amend the default links if you need to model restrictions of your network. The default links may be restored by running the command:

```
xCommand DefaultLinksAdd
```

8. Registration Control

The TANDBERG Gatekeeper can control which endpoints are allowed to register with it. Two separate mechanisms are provided: a simple Registration Restriction Policy, and an authentication process based on user names and passwords. It is possible to use both mechanisms at once: 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 Gatekeeper.

8.1. Setting Registration Restriction Policy

When an endpoint registers with your Gatekeeper it presents a list of aliases. You can control which endpoints are allowed to register by including any one of its aliases on the Allow List or the Deny list.

Entries on the Allow and Deny Lists are in the form of *patterns*. When an endpoint attempts to register, each of its aliases are compared with the patterns in the relevant list to see if they match. A pattern can either specify an exact alias, or use wildcards to specify a group of aliases whose registration you want to control.

For example, if the Registration Restriction policy is set to Deny 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, 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.

8.1.1. Viewing the Allow and Deny lists

To view the entries in the Allow and Deny lists, either issue the following commands:

```
xConfiguration Gatekeeper Registration AllowList
```

```
xConfiguration Gatekeeper Registration DenyList
```

or go to *Gatekeeper Configuration* -> *Restrictions*. The Allow and Deny list entries appear in the Allowed Registrations and Denied Registrations boxes respectively (see Figure 17).

8.1.2. Activating use of Allow or Deny lists

To activate the use of Allow or Deny lists when determining which aliases are allowed to register with the Gatekeeper, either issue the following command:

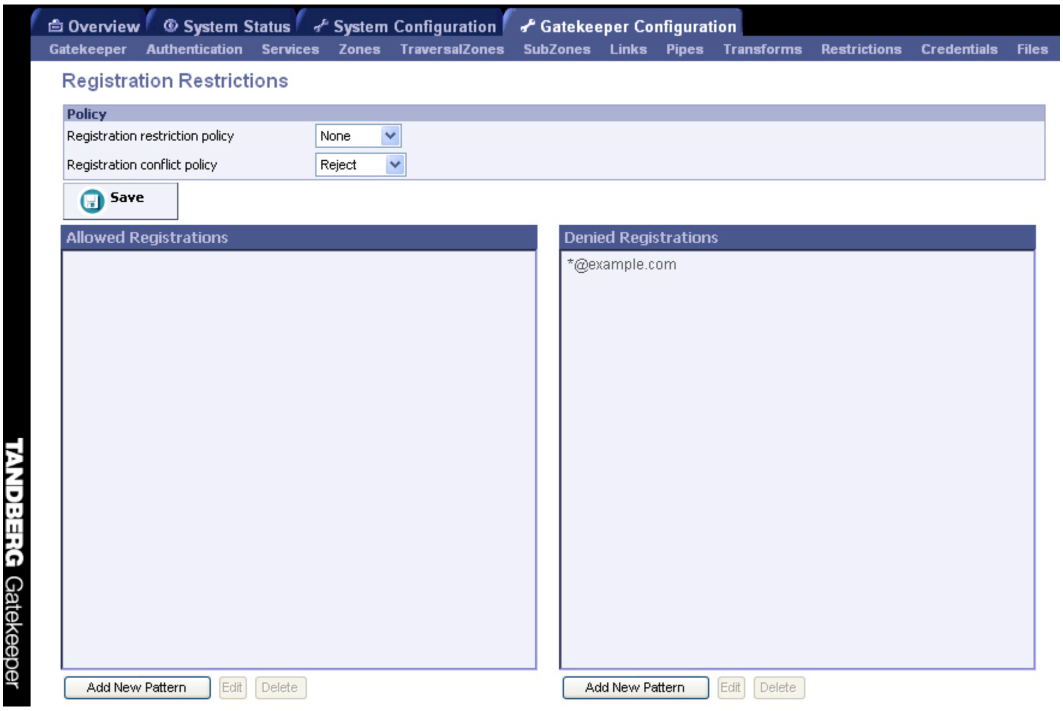
```
xConfiguration Gatekeeper Registration RestrictionPolicy
[None|AllowList|DenyList ]
```

or go to *Gatekeeper Configuration* > *Restrictions* and select one of the options from the Registration restriction policy drop-down menu.

The options are as follows:

`None` (default)	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.

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



Registration Restrictions

Policy

Registration restriction policy: None

Registration conflict policy: Reject

Allowed Registrations

Denied Registrations

*@example.com

Figure 17: Configuring registration restrictions

8.1.3. Managing entries in the Allow and Deny lists

When adding entries to the Allow and Deny lists, you can either specify an exact alias or use pattern matching to specify a group of aliases.

Pattern matching uses a simple form of wild card expansion:

wild card	definition
?	any single character
*	any single character or string of characters

For example:

Pattern	Match
12345678	Exact match only
1234567?	First 7 characters are an exact match, last character may be anything
123*	123 followed by anything
*@example.com	Anything ending with @example.com

To add and remove entries from the Allow and Deny lists, either issue the following commands:

```
xCommand AllowListAdd
xCommand AllowListDelete
xCommand DenyListAdd
xCommand DenyListDelete
```

or go to *Gatekeeper Configuration* > *Restrictions* and select Add New Pattern from underneath the appropriate list. In the Pattern field, enter the characters or pattern to be matched and select Save. The entry will now appear in the list. To edit or delete an existing pattern, highlight the pattern in the list and select either Edit or Delete.

8.2. Authentication

The TANDBERG Gatekeeper can use a user name and password based challenge-response scheme to permit registrations. For details of how to configure your endpoint with the appropriate information, please consult your endpoint manual.

The Gatekeeper supports the ITU H.235 specification [1] for authenticating the identity of network devices with which the Gatekeeper communicates.

In order to verify the identity of a device, the Gatekeeper needs access to the password information. This credential information may be stored in a local database on the Gatekeeper or obtained from an LDAP Directory Server.

Additionally, the Gatekeeper can be configured with its own username and password which it uses when authenticating with other systems, such as other TANDBERG Gatekeepers, Border Controllers or VCS systems.

8.2.1. Authentication and NTP

In order for an endpoint or other device to successfully authenticate with the Gatekeeper, the date and time on both systems must be synchronized. Accurate timestamps play an important part in authentication, helping to guard against replay attacks.

We recommend that all systems are synchronized through the use of an NTP server.

To configure the Gatekeeper with the details of an NTP server, either use the command line interface and issue the following command:

```
xconfiguration NTP Address: <IPAddress>
```

or use the web interface via *System Configuration* > *IP*, and in the Date and Time Settings section, enter the IP Address or name of the NTP server you wish to use.

8.2.2. Authentication using a local database

To configure the Gatekeeper to use the local database of credentials during authentication, either use the command line interface and issue the following commands:

```
xConfiguration Authentication Mode: On
xConfiguration Authentication Database: LocalDatabase
```

or use the web interface via *Gatekeeper Configuration* > *Authentication*, setting Authentication mode to `On` and the Authentication database to `LocalDatabase`.

Viewing credentials

To show the credentials in the local database, either use the command line interface and issue the following command:

```
xConfiguration Authentication Credential
```

or use the web interface via *Gatekeeper Configuration* > *Credentials.*

Managing credentials

Each credential in the local database has a username and a password. To manage the credentials in the local database, either use the command line interface to issue the following commands:

```
xcommand CredentialAdd <user name> <password>
xcommand CredentialDelete <credential index>
```

or use the web interface via *Gatekeeper Configuration* > *Credentials*. From here you can either add a new credential by selecting Add New Credential, or manage an existing credential by highlighting it and selecting Edit or Delete.

8.2.3. Authentication using an LDAP server

Authentication information can be obtained from an LDAP server. The directory on the LDAP server should be configured to implement the ITU H.350 specification [2] to store H.235 credentials for devices that the Gatekeeper communicates with. The directory should also be configured with the H.323 aliases of endpoints that will register with the Gatekeeper.

For instructions on how to configure common third party LDAP servers, see Appendix B.

To configure the Gatekeeper to use the LDAP server directory during authentication, either use the command line interface to issue the following commands:

```
xConfiguration Authentication Mode: On
xConfiguration Authentication Database: LDAPDatabase
```

or use the web interface via *Gatekeeper Configuration* > *Authentication*, setting Authentication mode to *On* and Authentication database to *LDAPDatabase*.

Configuring LDAP base DN

The Gatekeeper needs to be configured with the area of the directory which will be searched for the communication device information. This should be specified as the Distinguished Name (DN) in the directory under which the H.350 objects reside. To do this, either issue the following command:

```
xConfiguration Authentication LDAP BaseDN: "Your base DN"
```

or navigate to *Gatekeeper Configuration* > *Authentication* and enter the name of the directory in the LDAP base DN field.

Configuring LDAP server access

The Gatekeeper must also be configured with the location of the LDAP server and the security credentials required to gain access to the LDAP server.

To configure the LDAP server access, either issue the following commands:

```
xConfiguration LDAP Server Address: "ldap server address"
xConfiguration LDAP Server Port: 389
xConfiguration LDAP UserDN: "Your user DN"
xConfiguration LDAP Password: "password"
```

or navigate to *Gatekeeper Configuration* > *Authentication* and complete the relevant fields.

Viewing LDAP server connection status

To view the status of the connection between the Gatekeeper and the LDAP server, either issue the following command:

```
xstatus LDAP
```

or navigate to *Gatekeeper Configuration* > *Authentication*. The server status will be shown in a panel on the right-hand side of the screen.

8.2.4. Enforced dial plans

If LDAP authentication is in use, you may control what aliases an endpoint is allowed to register with. This allows you centralized control of your dial plan.

When an endpoint registers, it presents a list of aliases it wishes to use. You can control whether these aliases are used, replaced by those in the H.350 directory, or combined with those in the directory.

To set which aliases are used, either issue the following command:

```
xConfiguration Authentication LDAP AliasOrigin: <LDAP/Endpoint/Combined>
```

or navigate to *Gatekeeper Configuration* > *Authentication* and select the desired option from the LDAP alias origin drop-down menu.

The settings are as follows:

`LDAP` (default setting)	The LDAP aliases will be used and those presented by the endpoint ignored. If no aliases are present in the LDAP database for the endpoint which is registering, then the endpoint's aliases will be used.
`Endpoint`	The endpoint's aliases will be used, and any in the LDAP database will be ignored.
`Combined`	The endpoint will be registered with both the aliases which it has presented and those configured in the LDAP database.

8.2.5. Securing the LDAP connection with TLS

The traffic between the Gatekeeper and the LDAP server can be encrypted using Transport Layer Security (TLS). To use TLS, the LDAP server must have a valid certificate installed so that the Gatekeeper can verify the server's identity. For more information on setting up certificates using common LDAP servers, see Appendix B. LDAP uses port 636 as its default communications port.

To enable TLS, either issue the following command:

```
xConfiguration LDAP Encryption: TLS
```

or navigate to *Gatekeeper Configuration* > *Authentication* and from the LDAP Encryption drop-down menu select `TLS`.

The Gatekeeper will now only communicate with the LDAP server using TLS.

Uploading Trusted CA certificate

To verify the identity of the LDAP server, the certificate of the Certificate Authority (CA) that issued the LDAP server with its certificate must be uploaded to the Gatekeeper.

To install the CA's certificate, navigate to *Gatekeeper Configuration* > *Files* and upload the CA certificate as a Trusted CA certificate.

Note: Installation of the CA's certificate cannot be done via the command line interface.

8.2.6. Setting the Gatekeeper's own authentication credentials

You can configure the Gatekeeper with an authentication username and password that it can use when required to authenticate itself with other systems, for example if it is using the TANDBERG VCS Expressway as a traversal server. To set the authentication username and password, either use the command line interface to issue the following commands:

```
xConfiguration Authentication UserName: <S: 0, 25>
xConfiguration Authentication Password: <S: 0, 25>
```

or navigate to *Gatekeeper Configuration* > *Authentication* and in the External Registration Credentials section, enter the username and password to be used.

9. URI Dialing

9.1. About URI Dialing

If an alias is not located in the Gatekeeper's list of registrations, it may attempt to find an authoritative Gatekeeper through the DNS system.

URI dialing makes it easier for endpoints registered with different Gatekeepers or Border Controllers to call each other. Without URI dialing, you need to neighbor all the systems to each other. This does not scale well as the number of systems grows. It is also inconvenient for making one-off calls to endpoints registered with previously unknown systems.

Using URI dialing, you call using an H.323 URI which looks like an email address (e.g. john.smith@example.com). The destination Gatekeeper is found from the domain name -- the part after the @ -- in the same way that an email server is found.

9.2. Making a Call Using URI Dialing

9.2.1. Enabling URI dialing

To enable or disable URI dialing using the command line interface, issue the following command:

```
xConfiguration Gatekeeper DNSResolution Mode: <On/Off>
```

To enable or disable URI dialing using the web interface, navigate to *Gatekeeper Configuration* > *Gatekeeper* and tick or clear the Allow DNS resolution checkbox.

9.2.2. Configuring DNS server(s)

If URI dialing is enabled, you will also need to configure at least one DNS server for the systems to query. For resilience, you can specify up to five DNS servers.

To do this, either issue the following command:

```
xConfiguration IP DNS Server 1 Address: <address>
```

or navigate to *System Configuration* > *IP* and under the DNS section, enter the IP address(es) of the DNS server(s) you wish to use (see Figure 18).

Note: If you want others to be able to reach you using URI dialing, add a record to your DNS information as described in Appendix A

9.2.3. Configuring the domain name

We recommend that endpoints register with a full URI (e.g. firstname.lastname@example.com). However, your dial plan may be set up such that endpoints register with the Gatekeeper without their domain name (e.g. firstname.lastname). In this case, the Gatekeeper needs to match an incoming request for firstname.lastname@example.com to a registration for firstname.lastname. To do this, it must be configured with the name of the domain to which its endpoints belong.

To configure the domain name, either issue the following command:

```
xConfiguration Gatekeeper LocalDomain DomainName: <name>
```

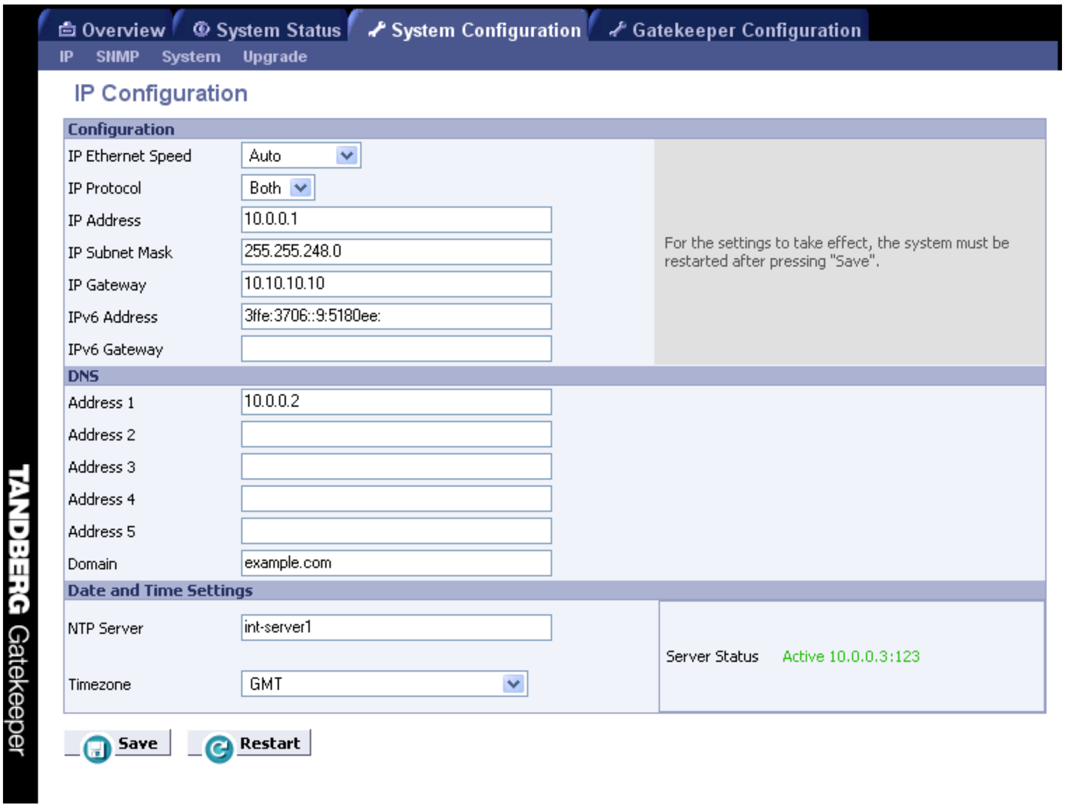
or navigate to *Gatekeeper Configuration* > *Gatekeeper* and in the Local Domain section in the Domain name field, enter the domain name.

The same local domain name should be set on both the Gatekeeper and the Border Controller. Any Alternates should also have the same local domain name.

9.2.4. URI dialing and firewall traversal

If URI dialing is being used in conjunction with firewall traversal, DNS Resolution should be enabled only on the Border Controller and any Gatekeepers on the public network. Gatekeepers behind the firewall should not have DNS resolution enabled.

In addition, the DNS records should be updated with the address of the Border Controller as the authoritative Gatekeeper for the enterprise (see Appendix A). This ensures that calls placed using URI dialing enter and leave the enterprise through the Border Controller, allowing successful traversal of the firewall.



IP Configuration

Configuration

IP Ethernet Speed	Auto
IP Protocol	Both
IP Address	10.0.0.1
IP Subnet Mask	255.255.248.0
IP Gateway	10.10.10.10
IPv6 Address	3ffe:3706::9:5180ee:
IPv6 Gateway	

For the settings to take effect, the system must be restarted after pressing "Save".

DNS

Address 1	10.0.0.2
Address 2	
Address 3	
Address 4	
Address 5	
Domain	example.com

Date and Time Settings

NTP Server	int-server1
Timezone	GMT

Server Status Active 10.0.0.3:123

Figure 18: Configuring IP interface

9.3. Receiving a Call Using URI Dialing

When an incoming call has been placed using URI dialing, the Gatekeeper will receive a request containing the dialed URI in the form `user@host`. As described in *DNS Records* (section 9.4), several mechanisms could have been used to locate the Gatekeeper. Depending on which was used, the received URI could be in one of three forms:

- user@10.0.0.1
- user@srv.record.domain.name
- user@a.record.domain.name

Each of these should be able to discover an endpoint registered as either *user* or *user@a.record.domain.name*.

On receipt of the URI the Gatekeeper will modify the URI by removing the @ and host if the host matches either:

- The IPv4 or IPv6 address of the Gatekeeper, or
- The system name of the system.

The Gatekeeper will then search for registrations which match either the modified URI, or the modified URI with its own Local Domain Domain Name appended.

9.4. DNS Records

URI dialing relies on the presence of records in the DNS information for the zone. For preference service (SRV) records should be used. These specify the location of a server for a particular protocol and domain. Their format is defined by an Internet standard (RFC 2782 [3]) as

```
_Service._Proto.Name TTL Class SRV Priority Weight Port Target
```

The Gatekeeper supports two types of SRV record as defined by H.323 Annex O. These are Location and Call, with `_Service` set to `_h323ls` and `_h323cs` respectively.

For URI dialing on a Gatekeeper `_Service` is defined by the H.323 protocol suite to be `_h323ls` and `_Proto` is _udp. `Name` corresponds to the host part of the H.323 URI.

Process

When the Gatekeeper receives a request to call `fred@example.com`, it will first search all the zones it knows about for that alias. If it can not be located, the Gatekeeper will then attempt to locate the destination using the DNS system, as follows:

1. First the Gatekeeper will query for a Location SRV record, to discover the authoritative Gatekeeper for the destination DNS zone.
2. If is not located, the Gatekeeper will query for a Call SRV record and try to place the call to that address.
3. If no appropriate SRV record can be located, the Gatekeeper will fall back to looking for an A or AAAA record for the domain. If a record is found, a call will be placed to that address.

If you intend to use URI dialing, you should provide at least a Location SRV record: it provides the most flexibility and the simplest configuration. 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.

Example configuration

Configuration of a system for a company with the domain name `example.com` might typically be:

- A record for `box.example.com` returns the IP address of the box
- SRV record for `_h323ls._udp.example.com` returns `box.example.com`
- SRV record for `_h323cs._tcp.example.com` returns `box.example.com`
- System name set to `box.example.com`
- `LocalDomain DomainName` set to `example.com`

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 Appendix A.

10. ENUM Dialing

10.1. About ENUM Dialing

ENUM provides another DNS-based dialing scheme. Users dial an E.164 number - a telephone number - which is converted in to an H.323 URI by the DNS system. The rules for URI dialing are then followed to place the call. This allows you to retain the flexibility of URI dialing whilst having the simplicity of calling using just a number.

Before the DNS lookup can be performed, the E.164 number must be transformed into a host name. To do this, the digits are reversed and separated by a dot -- similar to the way DNS PTR records are formed. The DNS zone is then appended.

For example, if an ENUM root of e164.example.com is being used, and the dialed number is +47 67 125 125, then the transformed host name is 5.2.1.5.2.1.7.6.7.4.e164.example.com

RFC 3761 [8], which defines the ENUM standard, specifies that the DNS zone for ENUM is e164.arpa. Use of this DNS zone requires that your E.164 numbers are assigned by an appropriate national regulatory body. Not all countries are yet participating in ENUM, so it may be useful to use an alternative DNS zone for ENUM. This could either be within your corporate DNS zone or could use a public ENUM database such as http://www.e164.org

The DNS zone used for ENUM contains NAPTR records as defined by RFC 2915 [7]. These provide the mapping between E.164 numbers and H.323 URIs.

The Gatekeeper may be configured with up to 5 DNS zones to search for a NAPTR record. It will iterate through them in order, stopping when the first record is found.

10.2. Configuring ENUM

10.2.1. Enabling ENUM support

ENUM support is disabled by default.

To enable ENUM support on your Gatekeeper, either:

enter the command:

```
xConfiguration Gatekeeper ENUM Mode: On
```

or navigate to *Gatekeeper Configuration* > *Gatekeeper* and in the ENUM section, tick the Allow ENUM resolution box.

10.2.2. Managing ENUM DNS zones

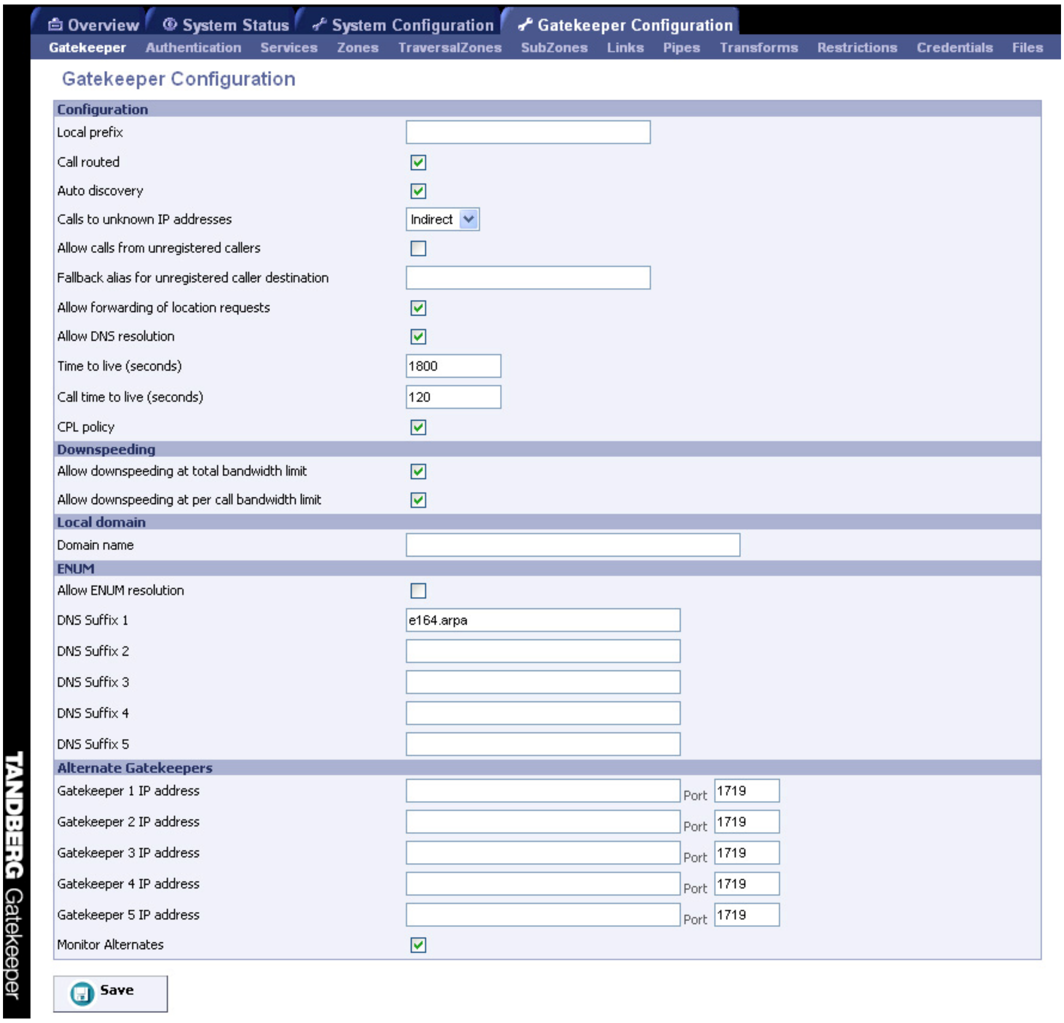
You are provided by default with the global ENUM DNS zone: e164.arpa. To change this or add other DNS zones, either:

enter the command:

```
xConfiguration Gatekeeper ENUM DNSSuffix [1..5 ]: <zone_name>
```

or navigate to *Gatekeeper Configuration* >*Gatekeeper* and in the ENUM section, enter details in the DNS Suffix... fields (see Figure 19).

Note: If you have a number of Gatekeepers and Border Controllers neighbored together, it is recommended that ENUM support is enabled on only one of them. If ENUM is enabled on more than one system, call set up could become unpredictable.



Gatekeeper Configuration

Configuration	
Local prefix	
Call routed	☑
Auto discovery	☑
Calls to unknown IP addresses	Indirect
Allow calls from unregistered callers	□
Fallback alias for unregistered caller destination	
Allow forwarding of location requests	☑
Allow DNS resolution	☑
Time to live (seconds)	1800
Call time to live (seconds)	120
CPL policy	☑
Downspeeding	
Allow downspeeding at total bandwidth limit	☑
Allow downspeeding at per call bandwidth limit	☑
Local domain	
Domain name	
ENUM	
Allow ENUM resolution	□
DNS Suffix 1	e164.arpa
DNS Suffix 2	
DNS Suffix 3	
DNS Suffix 4	
DNS Suffix 5	
Alternate Gatekeepers	
Gatekeeper 1 IP address	Port 1719
Gatekeeper 2 IP address	Port 1719
Gatekeeper 3 IP address	Port 1719
Gatekeeper 4 IP address	Port 1719
Gatekeeper 5 IP address	Port 1719
Monitor Alternates	☑

Figure 19: Setting the ENUM Zone

10.3. Configuring DNS NAPTR Records

ENUM relies on the presence of NAPTR records, as defined by RFC 2915 [7]. This is used to obtain an H.323 URI from the E.164 number. The record format that the Gatekeeper supports is:

```
;; order flag preference service regex replacement
IN NAPTR 10 100 "u" "E2U+h323" "!^(.*)$!h323:\1@example.com!" .
```

where:

`order(10)` and `preference(100)` determine the order in which NAPTR records will be processed: Lowest order first, with lowest preference being processed first in the case of matching order.

`flag (u)` determines the interpretation of the other fields in this record. Only the value *u* is supported.

`service` states that this record is intended to describe E.164 to URI conversion for H.323. Its value must be `E2U+h323`.

`regex` describes the conversion from the given E.164 number to an H.323 URI. `!` is a field separator. The first part: `^(.*)$` represents the entire E.164 number. The second part: `h323:\1@example.com` represents the H.323 URI that will be generated. In the above example the E.164 number will be concatenated with `@example.com`, e.g. `1234` will be mapped to `1234@example.com`.

The last field of the NAPTR record, `replacement`, is not used and should be set to `.` (i.e. the full stop character).

Once the DNS NAPTR (for the ENUM lookup) and SRV (for the corresponding H.323 URI lookup) are present, ENUM dialing should be possible. To verify your configuration, use the `locate` command to try and resolve an E.164 alias.

11. Example Traversal Deployments

11.1. Simple Enterprise Deployment

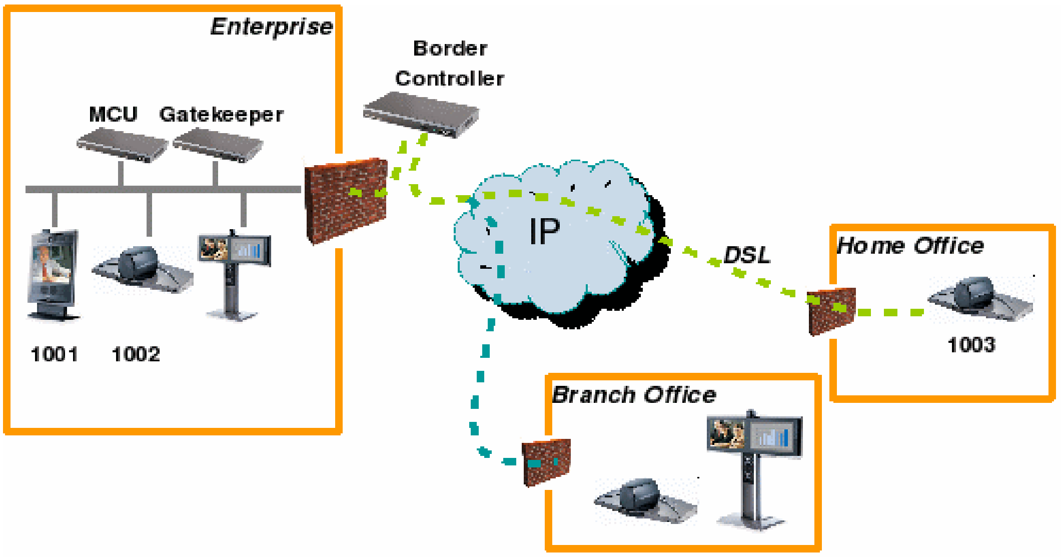


Figure 20: Simple enterprise deployment

Figure 20 shows a typical enterprise deployment. Endpoints 1001, 1002 and a Gatekeeper are deployed on a private network, separated from the public network by a firewall and NAT. Endpoint 1003 is on a separate private network, perhaps a home worker on an DSL connection. A Border Controller is deployed on the public network to allow traversal across the firewalls.

Endpoints 1001, 1002 may be any H.323-compliant endpoint. They will use the TANDBERG Gatekeeper to provide firewall traversal. Endpoint 1003 must be a TANDBERG endpoint that provides firewall traversal.

Endpoints 1001, 1002 should register with the Gatekeeper. Endpoint 1003 will register with the Border Controller. The Gatekeeper and Border Controller are configured to work together to provide firewall traversal.

11.2. Enterprise Gatekeepers

If your enterprise has already deployed a third-party Gatekeeper to manage calls within the private network, you may wish to deploy a traversal solution without having to alter the existing deployment.

In order to achieve this, the TANDBERG Gatekeeper is neighbored with the existing enterprise Gatekeeper as shown in Figure 21. The Enterprise Gatekeeper is also neighbored with the TANDBERG Gatekeeper.

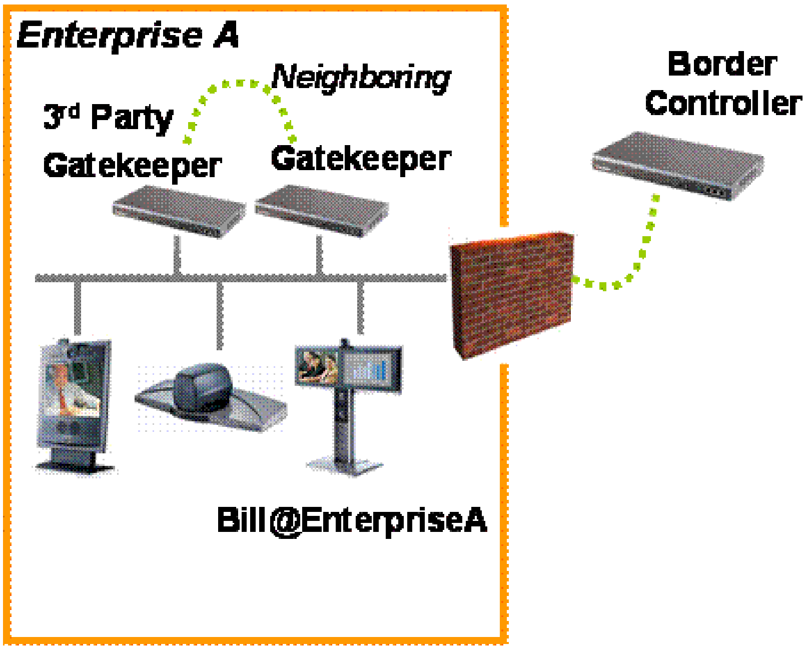


Figure 21: Neighboring with an enterprise gatekeeper

The TANDBERG Gatekeeper and Border Controller are configured as described in *Simple Enterprise deployment* (section 11.1), in order to provide firewall traversal.

11.3. Dialing Public IP Addresses

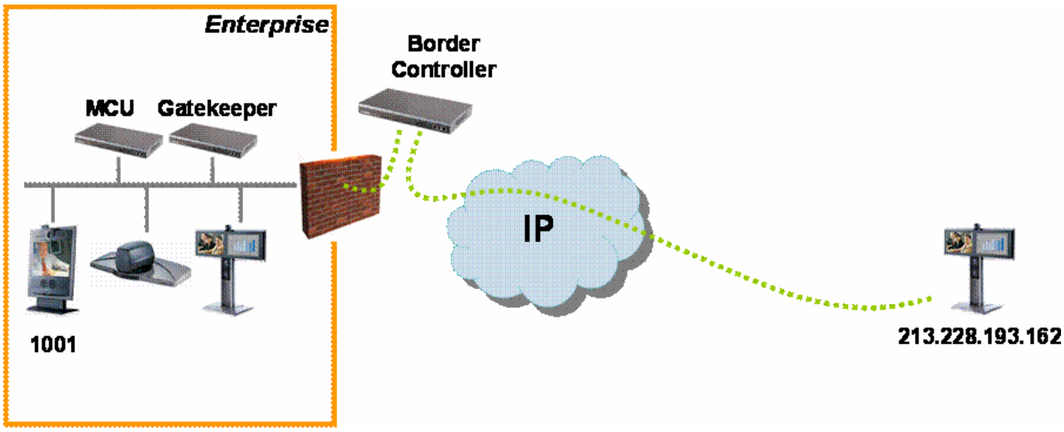


Figure 22: Dialing a public IP address

Figure 22 shows a private endpoint (1001) calling an endpoint on a public IP address. In this case the public endpoint is not registered to a Gatekeeper and can only be reached using its IP address. In order to successfully traverse the firewall it is necessary for the call to be relayed through the Border Controller; the TANDBERG Gatekeeper should not attempt to place the call directly to the public endpoint.

In order to achieve this:

1. On the Gatekeeper, set Calls to unknown IP addresses to Indirect. This can be done via either:

   ```
   xconfiguration Gatekeeper CallsToUnknownIPAddresses: Indirect
   ```

   or *Gatekeeper Configuration* -> *Gatekeeper* and in the Configuration section, from the Calls to unknown IP addresses drop-down menu selecting Indirect.

   This setting will mean that, when the Gatekeeper receives a call that was dialed using an IP address, it will forward the calls to the TANDBERG Border Controller, thereby allowing the Border Controller itself to relay the call to the endpoint on the public IP address. Note however that if the IP address received by the Gatekeeper matches that of any subzone configured on the Gatekeeper, it will attempt to place the call locally and will not forward it to the Border Controller.

2. On the Border Controller, set Calls to unknown IP addresses to Direct.

   This setting will allow the Border Controller to connect any call that it receives from the internal Gatekeeper out to systems on the public Internet.

3. From Endpoint 1001, dial 213.228.193.162.

11.4. Neighbored Enterprises

If two enterprises have deployed Border Controllers for firewall traversal, the two Border Controllers may be neighbored to allow calls to be placed from one enterprise to another. Neighboring will reduce call setup time compared to URI dialing (see *URI Dialing*, section 9). The disadvantage of neighboring is that the Border Controllers have to be configured with each other's addresses before the call can be made.

Each Gatekeeper and its matching Border Controller are neighbored as described in section 11.1. Border Controllers A and B are then neighbored together.

11.5. URI Dialing

11.5.1. Enabling outgoing URI calls

If you wish to enable users from within your enterprise to call a user in another enterprise using URI dialing, then the following configuration is required:

1. Enter the address of your DNS server on the Border Controller. This can be done via either:

   ```
   xConfiguration IP DNS Server Address: dns_server_ip_address
   ```

   or *System configuration* > *IP* and in the DNS section, entering the address in one of the Address fields.

2. Enable URI dialing on the Border Controller. This is because you want to use the Border Controller to resolve any H.323 URI received. This can be done via either:

   ```
   xConfiguration Gatekeeper DNSResolution Mode: On
   ```

   or *Border Controller Configuration* -> *Gatekeeper* and in the Configuration field, ticking the Allow DNS Resolution box.

3. Disable URI dialing on the Gatekeeper. This is because you wish calls to be routed from the private network to the Border Controller in order to traverse the firewall. This can be done via:

   ```
   xConfiguration Gatekeeper DNSResolution Mode: On
   ```

   or *Gatekeeper Configuration* -> *Gatekeeper* and in the Configuration field, ticking the Allow DNS Resolution box.

4. Configure the same local domain name on both the Gatekeeper and the Border Controller.

When an endpoint in your enterprise dials the full H.323 URI of an endpoint in another enterprise (for example, `Ben@EnterpriseB.com`), the call will be routed to your Border Controller. This will discover that Border Controller B is registered in DNS as responsible for Enterprise B, and will route the call to it. Border Controller B will receive the incoming call and route it accordingly.

URI dialing will send all queries for a particular domain to the same Border Controller. If you want to have URI dialing covering multiple Border Controllers, nominate one as the master. That system is registered in DNS and is set up with all the other Border Controllers and Gatekeepers as neighbors. When the master receives a URI dialing request for an endpoint it does not know about, it will query its neighbors.

11.5.2. Enabling incoming URI calls

In order to be able to receive calls placed to `example.com` using URI dialing, configure the following:

- Set `example.com` as the domain name you are using on both the Gatekeeper and Border Controller. This can be done via either:

  ```
  xConfiguration Gatekeeper LocalDomain DomainName: <name>
  ```

  or *Gatekeeper or Border Controller Configuration* -> *Gatekeeper* and in the Local Domain section in the Domain name field, enter the domain name.

- Update the DNS entry for `example.com` with an **A** record representing the Border Controller and an **SRV** record that returns the Border Controller's **A** record. See *DNS Records* (section 9.4) for details.

12. Third Party Call Control

12.1. About Third Party Call Control

The Gatekeeper provides a third party call control API which enables you to place calls, disconnect calls, or initiate a blind transfer of an existing call.

The API is provided through the command line interface; it is not available via the web interface.

12.2. Placing a Call

A call between two endpoints may be placed via the Gatekeeper by issuing the command:

```
xCommand Dial <aliasA> <aliasB>
```

where

A	the alias of the first endpoint
B	the alias of the second endpoint

This will return immediately and the Gatekeeper will attempt to place the call.

Like other asynchronous Gatekeeper commands, progress information may be obtained by registering for feedback using the command:

```
xFeedback Register status/calls
```

12.3. Transferring a Call

A call may be transferred using the Gatekeeper by issuing the command:

```
xCommand CallTransfer <call_index> <leg_index> <dest>
```

where:

call_index	the call to be transferred
leg_index	the endpoint to be disconnected
dest	the endpoint to which the call will be transferred

These indices may be determined through inspection of the output of `xStatus Calls`.

The Gatekeeper must be operating in call routed mode and call transfer must be enabled.

12.3.1. Enabling call routed mode

To enable call routed mode, either:

issue the command

```
xconfiguration Gatekeeper CallRouted: <On/Off>
```

or go to *Gatekeeper Configuration* -> *Gatekeeper* and in the Configuration section, tick the Call routed box.

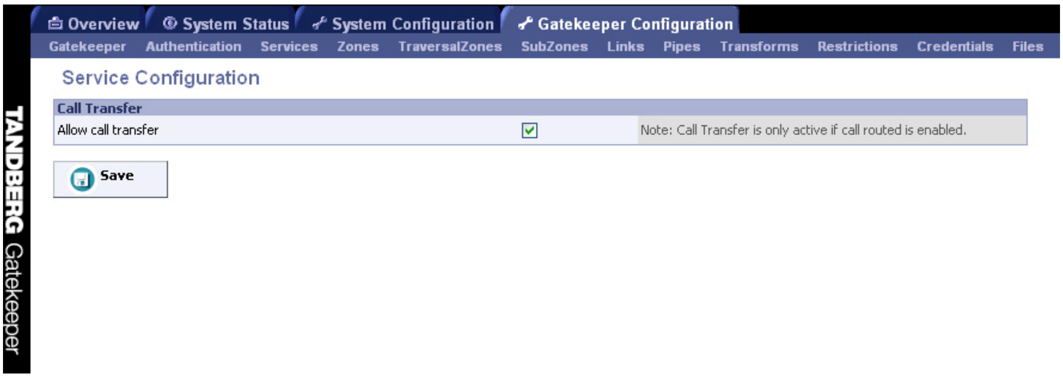
12.3.2. Enabling call transfer

To enable call transfer, either:

issue the command:

```
xConfiguration Services CallTransfer Mode: <On/Off>
```

or go to *Gatekeeper Configuration* -> *Services* and in the Call Transfer section, tick the Allow call transfer box (see Figure 23).



Service Configuration

Call Transfer		
Allow call transfer	☑	Note: Call Transfer is only active if call routed is enabled.

Figure 23: Enabling call transfer

12.4. Disconnecting a Call

An existing call may be disconnected using the Gatekeeper by issuing the command:

```
xCommand DisconnectCall: <index>
```

where:

`index`	the call index as reported by `xStatus Calls`

13. Multiway

13.1. About Multiway

Two callers in a simple point-to-point call may sometimes wish to bring other endpoints into the call. This can be achieved seamlessly using TANDBERG equipment, even if the two original endpoints do not support multi-site calling themselves. This functionality is known as “Multiway”.

13.2. Requirements

In order to enable Multiway calling, you must have the following:

- A TANDBERG endpoint that supports External Multisite, from which the ad hoc conference will be initiated.

Note: Other participants only need to be using an H.323-compliant endpoint that supports being put on hold.

- A TANDBERG MPS or Codian MCU to host the conference. The MPS/MCU must be configured with separate conference prefixes for both encrypted and non-encrypted conferences. It must also be configured with conference templates, represented by 3-digit codes Consult the MPS/MCU manual for details.

- A TANDBERG Gatekeeper to coordinate the conference transfer.

13.3. Process

During a normal point-to-point call, the user of the External Multisite-enabled endpoint can request that it be converted to a conference call. (Refer to the endpoint manual for details on how this is done.)

Upon receiving the ad hoc conference request from the endpoint, the Gatekeeper puts the two legs of the existing call on hold. It then creates a unique number that will be sent to the MPS/MCU as a conference request. This unique number is made up of:

- the ad hoc conference prefix. This indicates to the MPS/MCU the call type (currently encrypted or un-encrypted), and also defines which template it is to use for the conference.
  You must pre-configure this prefix on the Gatekeeper (see the following section for how this is done).

- the ad hoc conferencing ID. This tells the MPS/MCU which Gatekeeper has made the request. You must pre-configure this ID on the Gatekeeper (see the following section for how this is done).

- a unique suffix. This number is automatically assigned by the Gatekeeper and is different for each conference request.

The Gatekeeper then sends the conference request to the MPS/MCU. Upon receipt, the MPS/MCU creates a conference with the requested configuration. At this point the conference will have no calls.

Once the conference has been set up, the Gatekeeper routes each leg of the existing call to the conference. It also initiates an “on-hold” call between the new endpoint and the conference. Once all calls are established into the conference, they are all taken off hold and the participants will find themselves in the conference.

13.4. Configuration

1. Set Call Routed Mode to On. This can be done via either:

   ```
   xConfiguration Gatekeeper CallRouted: On
   ```

   or *Gatekeeper Configuration* > *Gatekeeper*. In the Configuration section, tick the Call Routed box.

2. Set Ad Hoc Conferencing Mode to On. This can be done via either:

   ```
   xConfiguration Services AdHocConferencing Mode: On
   ```

   or *Gatekeeper Configuration* > *Services*. In the Multiway section, tick the Allow ad hoc conferencing box.

3. Configure the Gatekeeper with the prefix it is to use for unencrypted conference requests. This must be a combination of two codes that have already been set up on the MPS/MCU, namely:

   - the call type code for unencrypted calls, followed by
   - the 3-digit code for the template to be used for the call.

   To set this prefix on the Gatekeeper, use either:

   ```
   xConfiguration Services AdHocConferencing Prefix: <S: 0, 30>
   ```

   or *Gatekeeper Configuration* > *Services*. In the Multiway section, enter the prefix in the Ad hoc conferencing prefix box.

Note: This prefix will be the same on all alternates registered to the MPS/MCU.

4. Configure the Gatekeeper with the prefix it is to use for encrypted conference requests. Again, this must be a combination of two codes that have already been set up on the MPS/MCU, namely:

   - the call type code for encrypted calls, followed by
   - the 3-digit code for the template to be used for the call.

   To set this prefix on the Gatekeeper, use either:

   ```
   xConfiguration Services AdHocConferencing Encryption Prefix:
   <S: 0, 30>
   ```

   or *Gatekeeper Configuration* > *Services*. In the Multiway section, enter the prefix in the Ad hoc conferencing encryption prefix box.

Note: This prefix will be the same on all alternates registered to the MPS/MCU.

1. Configure the Gatekeeper with a unique 3-digit ID. This ID is used by the MPS/MCU to distinguish between conference requests from different gatekeepers. Each gatekeeper and each Alternate that may initiate a Multiway conference on an MPS/MCU must use a different ID. To configure this ID, use either:

   ```
   xConfiguration Services AdHocConferencing ID: <S: 3, 3>
   ```

   or *Gatekeeper Configuration* > *Services*. In the Multiway section, enter the prefix in the Ad hoc conference identifier box.

14. Call Policy

14.1. About Call Policy

Your TANDBERG Gatekeeper allows you to set up policy to control which calls are allowed and even redirect selected calls to different destinations. You specify this policy by uploading a script written in the Call Processing Language (CPL). Each time a call is made the Gatekeeper executes the script 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
- Reject the call.

The Gatekeeper will only execute scripts for source or destinations which are registered directly with the system.

14.1.1. Uploading the CPL script

To upload the CPL script, go to *Gatekeeper Configuration* -> *Files.* In the Policy section, enter the path of the file in the CPL file field.

Note: The CPL script cannot be uploaded via the command line interface.

14.1.2. Enabling use of the CPL script

To enable or disable use of the CPL script, either:

issue the command:

```
xConfiguration Gatekeeper Policy Mode <On/Off>
```

or go to *Gatekeeper Configuration* -> *Gatekeeper* and in the Configuration section, tick or clear the CPL policy box.

14.1.3. Call Policy and Authentication

Policy interacts with authentication (see section 8.2). If authentication is enabled on the local Gatekeeper and a call is received from a remote, unauthenticated Gatekeeper, the call's source aliases will be removed from the call request before it is passed to the policy engine. This is because the unauthenticated source aliases could be forged and so should not be used for policy decisions in a secure environment.

14.1.4. CPL Standard

The following sections give details of the Gatekeeper's implementation of the CPL language and should be read in conjunction with the CPL standard (RFC 3880 [5]).

14.2. Making Decisions Based on Addresses

14.2.1. address-switch

The `address-switch` node allows the script to run different actions based on the source or destination aliases of the call. The `address-switch` specifies which fields to match and then a list of address nodes contains the possible matches and their associated actions.

The supported attributes on an `address-switch` and their interpretation are as follows:

`Field`

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
`origin`	The source aliases from the original LRQ or ARQ that started the call if it authenticated correctly otherwise `not-present`. Since SETUP messages are not authenticated, if we receive a SETUP without a preceding RAS message the origin will always be `not-present`.	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 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 source aliases from the original LRQ or ARQ that started the call if it authenticated correctly otherwise empty. Since SETUP messages are not authenticated if we receive a SETUP without a preceding RAS message the origin will always be `not-present`.	`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`	
`originating-zone`	The name of the zone or subzone for the originating leg of the call. If the call originates from a Zone or Traversal 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`
`destination`	The destination aliases.	
`original-destination`	The destination aliases.	

If the selected field contains multiple aliases then the Gatekeeper 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.

`subfield`

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.

<table>
<tr><td><code>address-type</code></td><td>For all alias types the address-type subfield is the string <code>h323</code></td></tr>
<tr><td><code>user</code></td><td>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.</td></tr>
<tr><td><code>host</code></td><td>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.</td></tr>
<tr><td><code>port</code></td><td>For IP addresses this is the port number in decimal.</td></tr>
<tr><td><code>tel</code></td><td>For E.164 numbers this selects the entire string of digits.</td></tr>
<tr><td><code>alias-type</code></td><td>Gives a string representation of the type of alias. The type is inferred from the format of the alias. Possible types are:
<table>
<tr><td>Address Type</td><td>Result</td></tr>
<tr><td>URI</td><td>url-ID</td></tr>
<tr><td>H.323 ID</td><td>h323-ID</td></tr>
<tr><td>Dialed Digits</td><td>dialedDigits</td></tr>
</table></td></tr>
<tr><td><code>display</code></td><td>Not defined for any alias types</td></tr>
</table>

address

The `address` construct is used within an address-switch to specify addresses to match. It supports the use of Regular Expressions (see *Appendix C* for further information).

Note: All address comparisons ignore upper/lower case differences so **`address is="Fred"`** will match **`fred`**, **`freD`** etc.

`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 Gatekeeper 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.

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 Gatekeeper 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 *CPL Examples*, section14.5).

14.3. CPL Script Actions

14.3.1. location

As the CPL script runs 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 `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 empty for incoming calls and to the original destination for outgoing calls.

The following attributes are supported on `location` nodes:

`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 (user@domain.com), H.323 ID or an E.164 number.

The `location` node supports the use of Regular Expressions (see *Appendix C* for further information).

14.3.2. proxy

On executing a `proxy` node the Gatekeeper will attempt to forward the call to the locations specified in the current location set. If multiple entries are in the location set then they are treated as different aliases for the same destination and are all placed in the destination alias field. If the current location set is empty the call will be forwarded to its original destination.

It is important to note that once a proxy node is executed, script execution stops immediately. There is currently no support for the proxy outputs `busy`, `noanswer` etc.

14.3.3. reject

If a `reject` node is executed the Gatekeeper stops any further script processing and rejects the current call.

14.4. Unsupported CPL Elements

The Gatekeeper 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 Gatekeeper 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

14.5. CPL Examples

14.5.1. Call screening of authenticated users

This example shows how to allow calls from only those users with authenticated source addresses.

See *Authentication* (section 8.2) for details on how to enable authentication.

```
<cpl>
  <incoming>
    <address-switch field="origin">
      <not-present>
        <reject/>
      </not-present>
    </address-switch>
  </incoming>
</cpl>
```

14.5.2. 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.

```
<cpl>
  <incoming>
    <address-switch field="destination">
      <address is="fred">
        <address-switch field="origin" subfield="host">
          <address subdomain-of="annoying.com">
            <reject/>
          </address>
          <otherwise>
            <proxy/>
          </otherwise>
          <not-present>
            <reject/>
          </not-present>
        </address-switch>
      </address>
    </address-switch>
  </incoming>
</cpl>
```

14.5.3. Call redirection

This example redirects all calls to user **barney** to voicemail.

```
<cpl>
  <incoming>
    <address-switch field="destination">
      <address is="barney">
        <location clear="yes" url="barney@voicemail">
          <proxy/>
        </location>
      </address>
      <otherwise>
        <proxy/>
      </otherwise>
    </address-switch>
  </incoming>
</cpl>
```

14.5.4. Call screening based on alias

In this example, user **ceo** will only accept calls from users **vpsales**, **vpmarketing** or **vpengineering**.

```
<cpl>
  <incoming>
    <address-switch field="destination">
      <address is="ceo">
        <address-switch field="origin">
          <address regex="vpsales|vpmarketing|vpengineering">
            <proxy/>
          </address>
          <otherwise>
            <reject/>
          </otherwise>
          <not-present>
            <reject/>
          </not-present>
        </address-switch>
      </address>
    </address-switch>
  </incoming>
</cpl>
```

14.5.5. Prevent external use of Gateway

In this example, we have an ISDN gateway registered with an alias of **MyGateway** that uses the prefixes of **0** and **9** to route outbound ISDN calls. The following script shows how to prevent callers from outside your network calling in via the gateway and then using it to make outbound calls.

```
<cpl>
  <incoming>
    <address-switch field="origin">
      <address is="MyGateway">
        <!-- check if this is an incoming call from the ISDN gateway -->
        <address-switch field="destination" subfield="tel">
          <address subdomain-of="0">
            <!--  rejects calls to aliases/services starting with 0 -->
            <reject status="reject" />
          </address>
          <address subdomain-of="9">
            <!--  rejects calls to aliases/services starting with 9 -->
            <reject status="reject" />
          </address>
        </address-switch>
      </address>
    </address-switch>
  </incoming>
</cpl>
```

15. Logging

15.1. About Logging

The Gatekeeper provides logging for troubleshooting and auditing purposes.

15.2. Viewing the event log

To view the event log, either issue the command:

```
eventlog [n/all]
```

where

`n`	The number of lines (from end of event log) to display.
`all`	Displays the whole event log.

or go to *System Status* -> *Event Log*.

15.3. Controlling what is Logged

15.3.1. About Event levels

All Events have an associated level in the range 1-3, with level 1 considered the most important. The table below shows the levels assigned to different events:

Level	Assigned Events
Level 1 (User)	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 (Protocol)	Logs of protocol messages sent and received (H.323, LDAP, etc) excluding noisy messages such as H.460.18 keep-alives and H.245 video fast-updates. . .
Level 3 (Protocol Verbose)	Protocol keep-alives are suppressed at Level 2. At logging level 3, keep-alives are also logged.

15.3.2. Setting the log level

You can control which events are logged by the Gatekeeper by specifying the log level. All events with a level numerically equal to and lower than the specified logging level are recorded in the event log.

To set the log level, either issue the command:

```
xConfiguration Log Level: <1..3>
```

or go to *System Configuration* -> *System* and in the Logging section, select the desired level from the Log Level drop-down menu.

By default, logging is set to level 1.

15.4. Event Log Format

The event log is displayed in an extension of the UNIX syslog format:

```
date time host_name facility_name <PID>: message_details
```

where

`date`	the local date on which the message was logged
`time`	the local time at which the message was logged
`host_name`	the name of the system generating the log message
`facility`	the name of the program generating the log message. This will be `gk` for all messages originating from TANDBERG Gatekeeper processes, but will differ for messages from third party processes which are used in the Gatekeeper product.
`message_details`	the body of the message (see below for further information)

For all messages logged from the `tandberg` process the field is structured to allow easy parsing. It consists of a number of human-readable *name=value* pairs, separated by a space. The first field is always:

Field	Example	Description
Event	`Event=RegistrationRequest`	The event which caused the log message to be generated.

and the last fields of the message are always:

Field	Example	Description
Level	`Level=1`	The level of the event being logged.
Time	`Time=2006/20/01-14:02:17`	The UTC date and time at which the event was generated.

15.5. Logged Events

Events logged at level 1

Event	Description
`Eventlog Cleared`	An operator cleared the event log.
`Admin Session Start`	An administrator has logged onto the system.
`Admin Session Finish`	An administrator has logged off the system.
`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.
`Policy Change`	A policy file has been updated.
`Registration Requested`	A registration has been requested.
`Registration Accepted`	A registration request has been accepted.
`Registration Rejected`	A registration request has been rejected. The `Reason` event parameter contains the H225 cause code. Optionally, the `Detail` event parameter may contain a textual representation of the H.225 additional cause code.
`Registration Removed`	A registration has been removed by the Gatekeeper/Border Controller. 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)
`Call Answer Attempted`	An attempt to answer a call has been made.
`Call Attempted`	A call has been attempted.
`Call Connected`	A call has been connected.
`Call Disconnected`	A call has been disconnected.
`Call Rejected`	A call has been rejected. The `Reason` event parameter contains a textual representation of the H.225 additional cause code.
`Call Bandwidth Changed`	The bandwidth of a call has changed.
`Decode Error`	A syntax error was encountered when decoding an H.323 message.

Event	Description
`External Server Communication Failure`	Communication with an external server failed unexpectedly. The event detail data should differentiate between 'no response' and 'request rejected'. Servers concerned are: • DNS servers • LDAP servers • Neighbor Gatekeepers • NTP servers
`System Start`	The operating system has started.
`System Shutdown`	The operating system was shutdown.
`Application Start`	The Gatekeeper has started. Further detail may be provided in the event data `Detail` field.
`Application Failed`	The Gatekeeper application is out of service due to an unexpected failure.
`License Limit Reached`	Licensing limits for a given feature have been reached. The event detail field specifies the facility/limits concerned. Possible values for the detail field are: • Non Traversal Call Limit Reached • Traversal Call Limit Reached • Registration Limit Reached
`Message Rejected`	The Authentication mode is set to On, and an endpoint has unsuccessfully attempted to send a message (such as a registration request) to the Gatekeeper. This could be either because the endpoint has not supplied any authentication credentials, or because its credentials do not match those expected by the Gatekeeper.

Events logged at level 2

Event	Description
`Message Received`	An incoming message has been received
`Message Sent`	An outgoing message has been sent

Event data fields

Each Event has associated data fields. Fields are listed below in the order in which they appear in the log message.

Field	Description	Applicable Events
`Protocol`	Specifies which protocol was used for the communication. Valid values are `TCP` or `UDP`	Call Attempted Call Bandwidth Changed Call Connected Call Disconnected Call Rejected External Server Communication Failure Message Sent Message Received Registration Accepted Registration Rejected Registration Removed Registration Requested
`Reason`	Textual string containing any reason information associated with an event.	Call Rejected External Server Communication Failure Registration Rejected Registration Removed
`Service`	Specifies which protocol was used for the communication. A service entry is one of: • H.225 • H.245 • NTP • DNS • LDAP • Neighbor Gatekeeper	External Server Communication Failure Message Sent Message Received
`Message Type`	Specifies the type of the message.	Message Sent Message Received

Field	Description	Applicable Events
`Src-ip`	Specifies the source IP address (the IP address of the device attempting to establish communications). The source IP is recorded in the dotted decimal format: (number).(number).(number).(number) or the IPv6 colon separated format.	Call Attempted Call Bandwidth Changed Call Connected Call Disconnected Call Rejected External Server Communication Failure Message Sent Message Received Registration Accepted Registration Rejected Registration Removed Registration Requested
`Dst-ip`	Specifies the 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`.	As `Src-ip`
`Dst-port`	Specifies the destination port: the IP port of the destination for a communication attempt.	As `Src-ip`
`Src-port`	Specifies the source port: the IP port of the device attempting to establish communications.	As `Src-ip`
`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	Registration Requested Call Attempted Call Connected Call Disconnected Call Rejected Call Bandwidth Changed Incoming Message* Outgoing 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	Registration Accepted Registration Removed Registration Rejected Call Attempted Call Connected Call Disconnected Call Rejected Message Sent* Message Received* Call Bandwidth Changed
`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.	All events

Field	Description	Applicable Events
`Level`	The level of the event as defined in section 15.3.1.	All events

* Included if event parameter relevant or available for message concerned.

In addition to the events described above, a `syslog.info` event containing the string `MARK` will be logged once an hour to provide confirmation that logging is still active.

15.6. Remote Logging

The event log is stored locally on the Gatekeeper. 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: A Gatekeeper will not act as a central logging server for other systems.

15.6.1. Enabling remote logging

To enable remote logging, the Gatekeeper must be configured with the address of the central log server. To do this, either issue the command:

```
xConfiguration Log Server Address: server_address
```

or go to *System Configuration* -> *System* and in the Logging section, enter the name of the server in the Remote Syslog Server field.

Note: Events will be always logged locally regardless of whether or not remote logging has been enabled.

16. Software Upgrading

16.1. About Software Upgrading

Software upgrade can be done in one of two ways:

- Using a web browser (HTTP/HTTPS).
- Using secure copy (SCP).

Note: To upgrade the Gatekeeper, a valid Release key and software file is required. Contact your TANDBERG representative for more information.

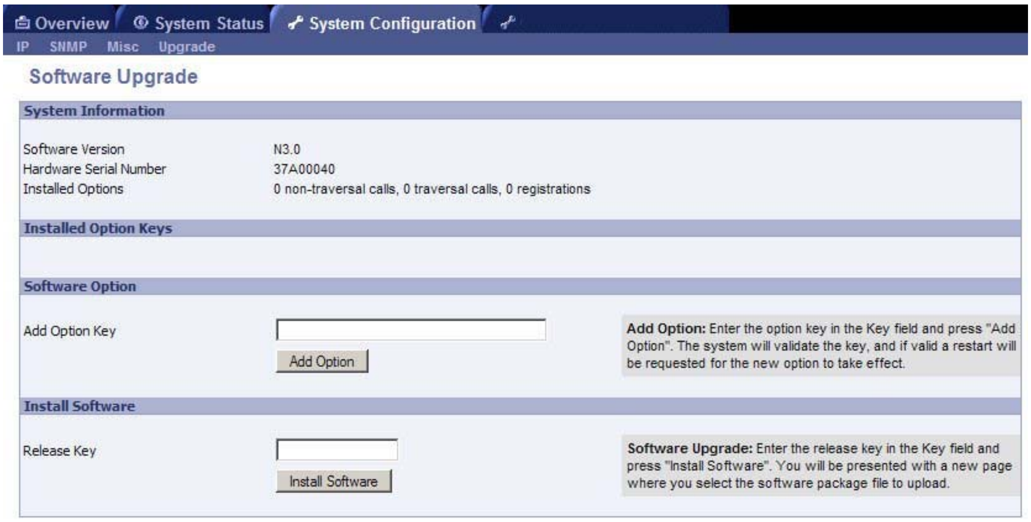
Note: Configuration is restored after performing an upgrade but we recommend that you make a backup of the existing configuration using the TANDBERG Management Suite before performing the upgrade.

16.2. Upgrading Using HTTP(S)

To upgrade using HTTP(S):

1. Go to *System Configuration* -> *Upgrade*.

   You will see the following screen:



Software Upgrade

System Information

Software Version	N3.0
Hardware Serial Number	37A00040
Installed Options	0 non-traversal calls, 0 traversal calls, 0 registrations

Installed Option Keys

Software Option

Add Option Key

Add Option

Add Option: Enter the option key in the Key field and press "Add Option". The system will validate the key, and if valid a restart will be requested for the new option to take effect.

Install Software

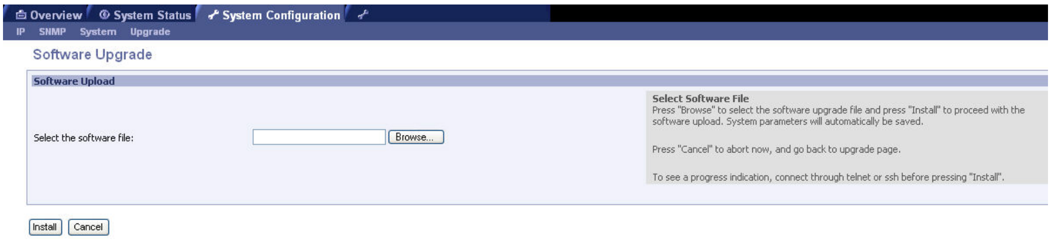
Release Key

Install Software

Software Upgrade: Enter the release key in the Key field and press "Install Software". You will be presented with a new page where you select the software package file to upload.

2. In the Install Software section, enter your key in the Release Key field and select Install Software.

   You will see the following screen:



Software Upgrade

Software Upload

Select the software file:

Browse...

Select Software File
Press "Browse" to select the software upgrade file and press "Install" to proceed with the software upload. System parameters will automatically be saved.

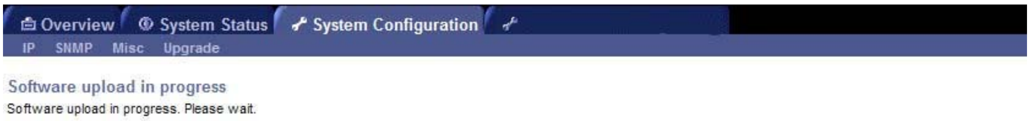
Press "Cancel" to abort now, and go back to upgrade page.

To see a progress indication, connect through telnet or ssh before pressing "Install".

Install Cancel

3. Browse to the file containing the software and select Install.

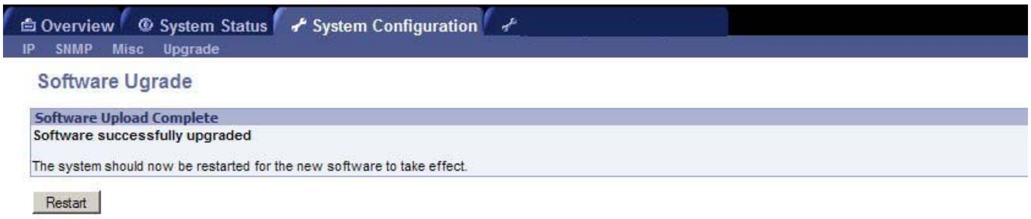
   You will see a page indicating that upload is in progress:



Software upload in progress

Software upload in progress. Please wait.

   When the upload is completed you will see the following:



Software Ugrade

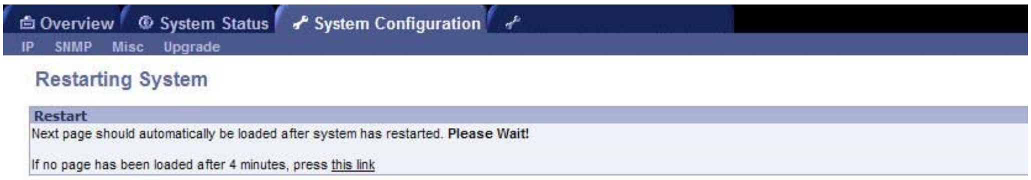
Software Upload Complete

Software successfully upgraded

The system should now be restarted for the new software to take effect.

4. Select Restart.

   You will see a confirmation window:



Restarting System

Restart

Next page should automatically be loaded after system has restarted. **Please Wait!**

If no page has been loaded after 4 minutes, press this link

   The system will then perform a second reboot to restore system parameters.

After 3-4 minutes, the Gatekeeper is ready for use.

16.3. Upgrading Using SCP/PSCP

To upgrade using SCP or PSCP (part of the PuTTY free Telnet/SSH package) you need to transfer two files to the Gatekeeper:

- a text file containing the release key, and
- a file containing the software image.

Note: Make sure you transfer the release key file before transferring the software image. Also make sure you name the files exactly as described below.

Note: The release key file should contain just the 16 character release key.

To upgrade using SCP or PSCP:

1. Make sure the system is turned on and available on IP.
2. Upload the release key file using SCP/PSCP to the `/tmp` folder on the system e.g.

   `scp release-key root@10.0.0.1:/tmp/release-key` or
   `pscp release-key root@10.0.0.1:/tmp/release-key`
3. Enter password when prompted.
4. Copy the software image using SCP/PSCP. The target name must be `/tmp/tandberg-image.tar.gz`, e.g.

   `scp s42000n60.tar.gz root@10.0.0.1:/tmp/tandberg-image.tar.gz` or
   `pscp s42000n60.tar.gz root@10.0.0.1:/tmp/tandbergimage.tar.gz`
5. Enter password when prompted.
6. Wait until the software has installed completely. This should not take more than two minutes.
7. Reboot the system.

After about four minutes the system will be ready to use.

17. Command Reference

This chapter lists the basic usage of each command. The commands also support more advanced usage, which is outside the scope of this document.

17.1. Status

The status root command, `xstatus`, returns status information from the Gatekeeper.

17.1.1. Listing all status information

To list all status information, type:

```
xstatus
```

Status is reported hierarchically beneath the status root. It is possible to reduce the amount of information returned by `xstatus` by specifying a more detailed status command.

17.1.2. Listing all status commands

To list all `xstatus` commands available at the root level type:

```
xstatus ?
```

17.1.3. Calls

`xstatus Calls`

Returns information about all active calls on the system.

`xstatus Calls Call <`*`index`*`>`

Returns information about the specified call.

17.1.4. Ethernet

`xstatus Ethernet`

Returns the currently active configuration of the Ethernet interface.

`MacAddress`	Returns the MAC address of the LAN 1 interface.
`Speed`	Returns the speed of the Ethernet link. Reports Down if the link is down or not connected.

17.1.5. ExternalManager

`xstatus ExternalManager`

Returns information about the external manager. The External Manager is the remote system, such as the TANDBERG Management Suite (TMS) used to manage the endpoints and network infrastructure.

`Address`	Returns the IP address of the external manager.
`Protocol`	Returns the Protocol used to communicate with the external manager.
`URL`	Returns the URL used to communicate with the external manager.

17.1.6. Feedback

`xstatus Feedback`

Returns all currently registered feedback expressions.

`xstatus Feedback <`*`index`*`>`

Returns the specified feedback expression.

17.1.7. IP

`xstatus IP`

Returns the active IP configuration of the system including protocol, IP address, subnet mask and gateway.

If you have changed the IP configuration without rebooting, `xstatus IP` will return the original settings currently in effect.

`Protocol`	Returns the Protocol in which the system is operating: IPv4, IPv6 or both
`Address`	Returns the system's IP address
`SubnetMask`	Returns the system's IP subnet mask
`Gateway`	Returns the system's IPv4 gateway
`V6`	Returns the system's IPv6 address and gateway
`DNS`	Returns the address(es) of the DNS servers in use, and the system's domain name

17.1.8. LDAP

`xstatus LDAP`

Returns the status of any connection to an LDAP server.

17.1.9. Links

xstatus Links

Reports call and bandwidth information for all links on the system.

xstatus Links Link *<index>*

Reports call and bandwidth information for the specified link.

`Name`	Returns the name assigned to this link
`Calls`	Returns a list of call indices for calls currently active on this link.
`Bandwidth`	Returns the total and per-call bandwidth limits on this link, together with bandwidth currently in use.

17.1.10. NTP

xstatus NTP

Reports the status of any connection to an NTP server.

17.1.11. Options

xstatus Options

Reports the status of the option keys installed on the system.

17.1.12. Pipes

xstatus Pipes

Returns call and bandwidth information for all pipes on the system.

xstatus Pipes Pipe *<index>*

Reports call and bandwidth information for the specified pipe.

17.1.13. Registrations

xstatus Registrations

Returns a list of all registered endpoints on the system and their information.

xstatus Registrations Registration *<index>*

Returns information about the specified registration.

17.1.14. ResourceUsage

`xstatus ResourceUsage`

Returns information about the usage of system resources.

`Registrations`	Number of currently active registrations.
`MaxRegistrations`	Maximum number of concurrent registrations since system
`TraversalCalls`	Number of currently active traversal calls.
`MaxTraversalCalls`	Maximum number of traversal calls since system start.
`TotalTraversalCalls`	Total number of traversal calls since system start.
`NonTraversalCalls`	Number of currently active non traversal calls.
`MaxNonTraversalCalls`	Maximum number of non traversal calls since system start.
`TotalNonTraversalCalls`	Total number of non traversal calls since system start.

17.1.15. SubZones

`xstatus SubZones`

Returns call and bandwidth information for all subzones on the system.

`xstatus SubZones SubZone <`*`index`*`>`

Returns call and bandwidth information for the specified subzone.

17.1.16. SystemUnit

`xstatus SystemUnit`

Reports information about the system as follows:

- Product name
- Uptime
- SystemTime
- TimeZone
- LocalTime
- Software version
- Software Build
- Software name
- Software release date
- Software release key
- Number of calls supported
- Number of registered endpoints and services supported
- Encryption supported
- Multiway supported

- Hardware serial number
- Hardware version

17.1.17. Zones

`xstatus Zones`

Returns call and bandwidth information for all zones on the system. Also shows status of the zone as a whole and the status of each gatekeeper in the zone.

17.2. Configuration

The configuration root command, `xconfiguration`, is used to configuration the system's settings.

To list all `xconfiguration` commands type:

```
xconfiguration ?
```

To list all configuration data, type:

```
xconfiguration
```

To list the data relating to a specific configuration command, type:

```
xconfiguration <command_name>
```

To show usage information for a specific configuration command, type:

```
xconfiguration <command_name> ?
```

To set a configuration element type:

```
xconfiguration <command_name> param1: value1 param2: value2
```

Note: Remember to use the colon after naming the parameters.

17.2.1. Authentication

The `Authentication` group of commands allow you to configure parameters relating to how an endpoint authenticates itself with the Gatekeeper.

`xconfiguration Authentication Credential [1..1000] Name: <username>`

Specifies the username of a credential in the local authentication database.

`xconfiguration Authentication Credential [1..1000] Password: <password>`

Specifies the password of a credential in the local authentication database.

`xconfiguration Authentication Database: <LocalDatabase/LDAPDatabase>`

Selects between a local database and a remote LDAP repository for the storage of password information for authentication. The default is `LocalDatabase`.

`xconfiguration Authentication LDAP BaseDN: <S: 0, 255>`

Specifies the Distinguished Name to use when connecting to an LDAP server. The default is an empty string.

`xconfiguration Authentication LDAP AliasOrigin: <LDAP/Endpoint/Combined>`

Specifies which aliases -- from the endpoint or the database -- should be used to register the endpoint. Defaults to `LDAP`.

xconfiguration Authentication Mode: <On/Off>

Specifies whether or not to use H.235 authentication of calls and registrations. The default is `Off`: no authentication is required.

xconfiguration Authentication UserName: <username>

Specifies the user name to be used by the Gatekeeper when authenticating with another system.

xonfiguration Authentication Password: <password>

Specifies the password to be used by the Gatekeeper when authenticating with another system.

17.2.2. Ethernet

xconfiguration Ethernet Speed: <Auto/10half/10full/100half/100full>

Sets the speed of the Ethernet link. Use `Auto` to automatically configure the speed. The default is `Auto`.

You must restart the system for changes to take effect.

Note: to find out the current speed, use `xstatus Ethernet Speed`.

17.2.3. ExternalManager

xconfiguration ExternalManager Address: <address>

Sets the address of the External Manager. This can be either an IP Address or, if DNS is configured, a FQDN. The External Manager is the remote system, such as the TANDBERG Management Suite (TMS), used to manage endpoints and network infrastructure.

xconfiguration ExternalManager Path: <path>

Sets the URL of the External Manager.

17.2.4. Gatekeeper

Commands under the `Gatekeeper` node control aspects of the system's operation as an H.323 gatekeeper.

xconfiguration Gatekeeper Alternates Monitor: <On/Off>

Controls whether or not alternate gatekeepers are periodically interrogated to ensure that they are still functioning. In order to prevent delays during call setup, non-functional alternates will not receive Location Requests .

xconfiguration Gatekeeper Alternates Alternate [1..5] Address: <*IPAddress*>

Sets the IP address of an alternate Gatekeeper. Up to 5 alternates may be configured. When the Gatekeeper receives a Location Request, all alternates will also be queried.

xconfiguration Gatekeeper Alternates Alternate [1..5] Port: <*Port*>

Sets the IP port of an alternate Gatekeeper. The default is `1719`.

xconfiguration Gatekeeper AutoDiscovery: <On/Off>

Specifies whether or not the Gatekeeper responds to gatekeeper discovery requests from endpoints. The default is `On`.

`xconfiguration Gatekeeper CallRouted: <On/Off>`

Specifies whether the Gatekeeper should operate in call routed mode. The defaults is `Off`.

xconfiguration Gatekeeper CallsToUnknownIPAddresses: <Off/Direct/Indirect>

Specifies whether or not the Gatekeeper will attempt to call systems which are not registered with it or one of its neighbor gatekeepers. Options are:

`Direct`	Allows an endpoint to make a call to an unknown IP address without the Gatekeeper querying any neighbors. The call setup would occur just as it would if the far end were registered directly to the local system.
`Indirect` (default)	Upon receiving a call to an unknown IP address, the Gatekeeper will query its neighbors for the remote address, relying on the response from the neighbor to allow the ability for the call to be completed; connecting through the routing rules as it would through the neighbor relationship.
`Off`:	This will not allow any endpoint registered directly to the Gatekeeper to call an IP address of any system not also registered directly to that Gatekeeper.

See *Unregistered Endpoints* (section 6) for further detail. The default is `Indirect`.

xconfiguration Gatekeeper CallTimeToLive: <60..65534>

Specifies the interval in seconds at which endpoints are polled to verify that they are still in a call. The default is `120` seconds.

xconfiguration Gatekeeper DNSResolution Mode: <On/Off>

Determines whether or not DNS lookup of H.323 URIs is enabled on this system. The default is `On`.

xconfiguration Gatekeeper Downspeed PerCall Mode: <On/Off>

Determines whether or not the system will attempt to downspeed a call if there is insufficient per-call bandwidth configured to fulfill the request. The default is `On`.

xconfiguration Gatekeeper Downspeed Total Mode: <On/Off>

Determines whether or not the system will attempt to downspeed a call if there is insufficient total bandwidth available to fulfill the request. The default is `On`.

xConfiguration Gatekeeper ENUM Mode: <On/Off>

Determines whether or not the Gatekeeper supports ENUM dialing.

xConfiguration Gatekeeper ENUM DNSSuffix [1..5]: <S: 0, 128>

When ENUM dialing is enabled, specifies a DNS zone to be appended to the transformed E.164 number to create an ENUM host name, for which DNS is then queried.

xconfiguration Gatekeeper ForwardLocationRequests: <On/Off>

Determines behavior on receipt of a location request (LRQ) from another Gatekeeper. If set to `On`, the Gatekeeper will first try to resolve the request locally. If it cannot, the request will be forwarded to neighbor Gatekeepers. The default is `On`.

xconfiguration Gatekeeper LocalDomain DomainName: <*name*>

Specifies the DNS name of the domain that the Gatekeeper is responsible for. Used when searching for matching endpoint registrations.

xconfiguration Gatekeeper LocalPrefix: <*prefix*>

Sets the local zone prefix of the system.

`xconfiguration Gatekeeper Policy Mode: <On/Off>`

Determines whether or not the CPL policy engine is active. The default is `On`.

xconfiguration Gatekeeper Registration AllowList [1..1000] Pattern: <*pattern*>

Specifies a pattern in the list of allowed registrations. If one of an endpoint's aliases matches one of the patterns in the Allow List, the registration will be allowed.

xconfiguration Gatekeeper Registration ConflictMode: <Overwrite/Reject>

Determines how the Gatekeeper will behave if an endpoint attempts to register aliases currently registered from another IP address. The default is `Reject`.

xconfiguration Gatekeeper Registration DenyList [1..1000] Pattern: <*pattern*>

Specifies a pattern in list of denied registrations. If one of an endpoint's aliases matches one of the patterns in the Deny List the registration will be denied.

xconfiguration Gatekeeper Registration RestrictionPolicy: <None/AllowList/DenyList>

Specifies the policy in use to determine who may register with the system. The default is `None`.

xconfiguration Gatekeeper TimeToLive: <60..65534>

Specifies the interval at which the system polls the endpoint in order to verify that it is still functioning. Specified in seconds. The default is `1800` seconds.

xconfiguration Gatekeeper Transform [1..200] Pattern: <S: 0, 60>

Specifies the pattern to be used when deciding whether or not to transform a destination alias.

xconfiguration Gatekeeper Transform [1..200] Priority: <1..65534>

Determines the order in which transforms are matched. The priority must be unique for each transform.

xconfiguration Gatekeeper Transform [1..200] Type: <Prefix/Suffix/Regex>

`Prefix/Suffix` determines whether the pattern string being checked should appear at the beginning or end of an alias. Alternatively, `Regex` indicates that the pattern string should be treated as a regular expression when matching.

xconfiguration Gatekeeper Transform [1..200] Behavior: <Strip/Replace>

Determines how to modify the matched part of the alias. If set to `Strip`, the matching prefix or suffix will removed from the alias. If set to `Replace`, the matching part of the alias will be substituted for the replace text. Note that `Strip` is not a supported option if the pattern type is set to `Regex`.

xconfiguration Gatekeeper Transform [1..200] Replace: <S: 0, 60>

Specifies the string to be used as a substitution for the part of the alias that matched the pattern.

xconfiguration Gatekeeper Unregistered Caller Mode: <on/off>

Specifies whether calls may be made by an unregistered endpoint. Defaults to `Off`.

xconfiguration Gatekeeper Unregistered Caller Fallback: <*alias*>

Specifies the alias to which calls are placed if the Gatekeeper receives a call setup containing no alias information.

17.2.5. HTTP/HTTPS

Commands under the `HTTP` and `HTTPS` nodes control web access to the Gatekeeper.

`xConfiguration HTTP Mode: <On/Off>`

Enables/disables HTTP support. The default is `On`.

You must restart the system for changes to take effect.

`xconfiguration HTTPS Mode: <On/Off>`

Enables/disables HTTPS support. The default is `On`.

You must restart the system for changes to take effect.

Note: If web access is required, we recommend that you enable HTTPS and disable HTTP for improved security.

17.2.6. IP

Commands under the `IP` node allow you to configure IP-related parameters. The TANDBERG Gatekeeper may be configured to use either IPv4 or IPv6 or both.

When entering IPv4 addresses, dotted quad notation is used: `127.0.0.1`.

When using IPv6, addresses are entered in colon hexadecimal form: `2001:db8::2AA:FF:FE9A:4CA2`.

`xConfiguration IPProtocol: <Both/IPv4/IPv6>`

Selects whether the Gatekeeper is operating in IPv4, IPv6 or dual stack mode.

`xconfiguration IP Address: <`*`IPAddress`*`>`

Specifies the IPv4 address of the system.

`xconfiguration IP SubnetMask: <`*`IPAddress`*`>`

Specifies the IPv4 subnet mask of the system.

`xconfiguration IP Gateway: <`*`IPAddress`*`>`

Specifies the IPv4 gateway of the system.

`xconfiguration IP V6 Address: <`*`IPAddress`*`>`

Specifies the IPv6 address of the system.

`xconfiguration IP V6 Gateway: <`*`IPAddress`*`>`

Specifies the IPv6 gateway of the system.

`xconfiguration IP DNS Server [1..5] Address: <`*`IPAddress`*`>`

Sets the IP address of the DNS servers to be used when resolving domain names. Normally only the first DNS server will be queried for address resolution. If it fails to respond, all DNS servers will be queried.

Note: All the IP commands listed above require a system restart before they take effect.

xconfiguration IP DNS Domain Name: <*name*>

Specifies the name to be appended to the domain name before a query to the DNS server is executed, when attempting to resolve a domain name which is not fully qualified.

Note: This parameter is only used when attempting to resolve server addresses such as LDAP servers, NTP servers etc. It plays no part in URI dialing: (see `xconfiguration gatekeeper localdomain`).

17.2.7. LDAP

Parameters under the `LDAP` node control the Gatekeeper's communication with an LDAP server.

xconfiguration LDAP Encryption: <Off/TLS>

Sets the encryption mode to be used on the connection to the LDAP server. The default is `Off`.

xconfiguration LDAP Password: <*password*>

Sets the password to be used when binding to the LDAP server.

xconfiguration LDAP Server Address: <*address*>

Sets the IP address of the LDAP server to be used when making LDAP queries. . This can be either an IP Address or, if DNS is configured, a FQDN.

xconfiguration LDAP Server Port: <1..65534>

Sets the IP port of the LDAP server to be used when making LDAP queries.

xconfiguration LDAP UserDN: <*userDN*>

Sets the user distinguished name to be used when binding to the LDAP server.

17.2.8. Links

xconfiguration Links Link [1..100] Name: <*linkname*>

Specifies the name of a link in the list of links.

xconfiguration Links Link [1..100] Node1 Name: <*nodename*>

Specifies the first node of a link. A node name may be either a Zone name or a SubZone name.

xconfiguration Links Link [1..100] Node2 Name: <*nodename*>

Specifies the second node of a link. A node name may be either a Zone name or a SubZone name.

xconfiguration Links Link [1..100] Pipe1 Name: <*pipename*>

Specifies the first pipe associated with a link.

xconfiguration Links Link [1..100] Pipe2 Name: <*pipename*>

Specifies the second pipe associated with a link.

17.2.9. Log

xConfiguration Log Level: <1..3>

Controls the granularity of event logging with 1 being the least verbose, 3 the most.

17.2.10. NTP

xconfiguration NTP Address: <*address*>

Sets the IP address of the NTP server to be used when synchronizing system time. This can be either an IP Address or, if DNS is configured, a FQDN.

Accurate timestamps play an important part in authentication, helping to guard against replay attacks.

17.2.11. Option Key

xConfiguration Option [1..64] Key: <*optionkey*>

Specifies the option key of your software option.

An Option Key/software option is added to the system in order to add extra functionality, such as increasing the system's capacity. Contact your TANDBERG representative for further information.

You must restart the system for changes to take effect.

Note: The command `xstatus SystemUnit Software Configuration` can be used to discover the existing options.

17.2.12. Pipes

xconfiguration Pipes Pipe [1..100] Bandwidth Total Limit: <1..100000000>

Bandwidth associated with a pipe, keyed by index.

xconfiguration Pipes Pipe [1..100] Bandwidth Total Mode: <None/Limited/Unlimited>

Whether or not a given pipe is enforcing total bandwidth restrictions. `None` corresponds to no bandwidth available.

xconfiguration Pipes Pipe [1..100] Bandwidth PerCall Limit: <1..100000000>

Per call bandwidth of a pipe.

xconfiguration Pipes Pipe [1..100] Bandwidth PerCall Mode: <None/Limited/Unlimited>

Whether or not a given pipe is enforcing per-call bandwidth restrictions. `None` corresponds to no bandwidth available.

xconfiguration Pipes Pipe [1..100] Name: <*pipename*>

Name for a pipe.

17.2.13. Services

xconfiguration Services AdHocConferencing Mode: <On/Off>

Controls whether or not Multiway is enabled.

xconfiguration Services AdHocConferencing ID: <id>

Specifies the unique 3-digit ID for this system. This ID is used by the MPS to distinguish between conference requests from different gatekeepers.

`xconfiguration Services AdHocConferencing Prefix: <prefix>`

Specifies the prefix to be used for unencrypted conference requests.

`xconfiguration Services AdHocConferencing Encryption Prefix: <prefix>`

Specifies the prefix to be used for encrypted conference requests.

`xConfiguration Services CallTransfer Mode: <On/Off>`

Controls whether or not third party call transfer is enabled. The Gatekeeper must also be operating in call routed mode.

17.2.14. Session

xconfiguration Session TimeOut: <0..65534>

Controls how long 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. The default is `0`. You must restart the system for changes to take effect.

17.2.15. SNMP

xconfiguration SNMP CommunityName: <*name*>

SNMP Community names are used to authenticate SNMP requests. SNMP requests must have this 'password' in order to receive a response from the SNMP agent in the Gatekeeper.

xconfiguration SNMP Mode: <On/Off>

Turn on/off SNMP support.

xconfiguration SNMP SystemContact: <*name*>

Used to identify the system contact via SNMP tools such as TANDBERG Management Suite or HP OpenView.

xconfiguration SNMP SystemLocation: <*name*>

Used to identify the system location via SNMP tools such as TANDBERG Management Suite or HP OpenView.

17.2.16. SSH

xconfiguration SSH Mode: <On/Off>

Enables/disables SSH and SCP support. You must restart the system for changes to take effect.

17.2.17. Subzones

xconfiguration SubZones DefaultSubZone Bandwidth PerCall Limit: <1..100000000>

Per call bandwidth of the default subzone.

xconfiguration SubZones DefaultSubZone Bandwidth PerCall Mode: <None/Limited/Unlimited>

Whether or not the default subzone is enforcing total bandwidth restrictions. `None` corresponds to no bandwidth available.

xconfiguration SubZones DefaultSubZone Bandwidth Total Limit: <1..100000000>

Total bandwidth available on the default subzone.

xconfiguration SubZones DefaultSubZone Bandwidth Total Mode: <None/Limited/Unlimited>

Whether or not the default subzone is enforcing per-call bandwidth restrictions. `None` corresponds to no bandwidth available.

xconfiguration SubZones TraversalSubZone Bandwidth PerCall Limit: <1..100000000>

Per-call bandwidth available on the traversal subzone.

xconfiguration SubZones TraversalSubZone Bandwidth PerCall Mode: <None/Limited/Unlimited>

Whether or not the traversal subzone is enforcing per-call bandwidth restrictions. `None` corresponds to no bandwidth available.

xconfiguration SubZones TraversalSubZone Bandwidth Total Limit: <1..100000000>

Total bandwidth available on the traversal subzone.

xconfiguration SubZones TraversalSubZone Bandwidth Total Mode: <None/Limited/Unlimited>

Whether or not the traversal subzone is enforcing total bandwidth restrictions. `None` corresponds to no bandwidth available.

xconfiguration SubZones SubZone [1..100] Bandwidth PerCall Limit: <1..100000000>

Per-call bandwidth available on the indexed subzone.

xconfiguration SubZones SubZone [1..100] Bandwidth PerCall Mode: <None/Limited/Unlimited>

Whether or not the indexed subzone is enforcing per-call bandwidth restrictions. `None` corresponds to no bandwidth available.

xconfiguration SubZones SubZone [1..100] Bandwidth Total Limit: <1..100000000>

Total bandwidth available on the indexed subzone.

xconfiguration SubZones SubZone [1..100] Bandwidth Total Mode: <None/Limited/Unlimited>

Whether or not the indexed subzone is enforcing total bandwidth restrictions. `None` corresponds to no bandwidth available.

xconfiguration SubZones SubZone [1..100] Name: <*subzonename*>

Name of the indexed subzone.

xconfiguration SubZones SubZone [1..100] Subnet [1-5] IP Address: <*IPAddr*>

IP address used (in conjunction with the `IP PrefixLength` command) to identify a subnet to be assigned to this subzone.

xconfiguration SubZones SubZone [1..100] Subnet [1-5] IP PrefixLength: <*length*>

Number of bits of the `Subnet IP Address` which must match for an IP address to belong in this subzone.

17.2.18. SystemUnit

`xconfiguration SystemUnit Name: <name>`

The name of the unit. Choose a name that uniquely identifies the system.

`xconfiguration SystemUnit Password: <password>`

Specify the password of the unit. The password is used to login with Telnet, HTTP(S), SSH, SCP, and on the serial port.

Note: To set an empty password type
`xconfiguration SystemUnit Password: ""`

17.2.19. Telnet

`xconfiguration Telnet Mode: <On/Off>`

Enables/disables Telnet support.

You must restart the system for changes to take effect.

Note: For secure operation you should use SSH in preference to Telnet.

17.2.20. TimeZone

`xconfiguration TimeZone Name: <`*`timezone_name`*`>`

Sets the local time zone. Time zone names follow the POSIX naming convention e.g. Europe/London or America/New_York.

17.2.21. Traversal

`xconfiguration Traversal Registration RetryInterval: <1..65534>`

Sets the interval in seconds at which the Gatekeeper will attempt to register with the Border Controller if its initial registration fails for some reason. The default is `120` seconds.

`xconfiguration Traversal AllowMediaDirect: <On/Off>`

Determines whether endpoints must route their media through the Gatekeeper or may, if capable, send media directly to the Border Controller.

17.2.22. Zones

Traversal zones control how the Gatekeeper communicates with a Border Controller with which it is cooperating to provide firewall traversal.

`xconfiguration Zones TraversalZone [1..50] Name: <`*`name`*`>`

Sets the name of the traversal zone.

`xconfiguration Zones TraversalZone [1..50] Gatekeeper Address: <Address>`

Sets the IP address of the Border Controller that this system will register with. This could be an IP address or, if DNS is configured, a FQDN.

`xconfiguration Zones TraversalZone [1..50] Gatekeeper HopCount: <1..255>`

Specifies the hop count to be used when originating an LRQ.

xconfiguration Zones TraversalZone [1..50] Match [1..5] Mode: <AlwaysMatch/PatternMatch/Disabled>

The zone match mode determines when an LRQ will be sent to gatekeepers in the zone. If the mode is set to `AlwaysMatch` the zone will always be queried. If the mode is set to `PatternMatch`, the zone will only be queried if the alias queried for matches the corresponding pattern. If the mode is set to `Disabled` the zone will never be queried.

xconfiguration Zones TraversalZone [1..50] Match [1..5] Pattern String: <*pattern*>

The pattern to be used when deciding whether or not to query a zone. This is only used if the zone's match mode is set to `PatternMatch`.

xconfiguration Zones TraversalZone [1..50] Match [1..5] Pattern Type: <Prefix/Suffix/Regex>

`Prefix/Suffix` determines whether the pattern string being checked should appear at the beginning or end of an alias. Alternatively, `Regex` indicates that the pattern string should be treated as a regular expression when matching.

xconfiguration Zones TraversalZone [1..50] Match [1..5] Pattern Behavior: <Strip/Leave/Replace>

Determines whether the matched part of the alias should be modified before an LRQ is sent to the indicated zone. If set to `Leave`, the alias will be unmodified. If set to `Strip`, the matching prefix or suffix will removed from the alias. If set to `Replace`, the matching part of the alias will be substituted for the replace text. Note that `Strip` is not a supported option if the pattern type is set to `Regex`.

xconfiguration Zones TraversalZone [1..50] Match [1..5] Pattern Replace: <S:0, 60>

The string to be used as a substitution for the part of the alias that matched the pattern.

xconfiguration Zones Zone [1..100] Name: <*name*>

Sets an administrator-specified name for the zone.

xconfiguration Zones Zone [1..100] Gatekeeper [1..6] Address: <*Address*>

Specifies the addresses of the gatekeepers in the zone. . These can be either an IP Address or, if DNS is configured, a FQDN. Multiple addresses allows support for alternate gatekeepers.

xconfiguration Zones Zone [1..100] Gatekeeper [1..6] Port: <*port*>

Specifies the port on which the indexed gatekeeper is listening for RAS messages.

xconfiguration Zones Zone [1..100] HopCount: <*count*>

Specifies the hop count to be used when originating an LRQ.

xconfiguration Zones Zone [1..100] Monitor: <On/Off>

If zone monitoring is enabled, an LRQ will be periodically sent to the zone gatekeeper. If it fails to respond, that gatekeeper will be marked as inactive.

xconfiguration Zones Zone [1..100] Match [1..5] Mode: <AlwaysMatch/PatternMatch/Disabled>

The zone match mode determines when an LRQ will be sent to gatekeepers in the zone. If the mode is set to `AlwaysMatch` the zone will always be queried. If the mode is set to `PatternMatch`, the zone will only be queried if the alias queried for matches the corresponding pattern. If the mode is set to `Disabled` the zone will never be queried.

`xconfiguration Zones Zone [1..100] Match [1..5] Pattern String: <`*`pattern`*`>`

The pattern to be used when deciding whether or not to query a zone. This is only used if the zone's match mode is set to `AlwaysMatch`.

`xconfiguration Zones Zone [1..100] Match [1..5] Pattern Type: <Prefix/Suffix/Regex>`

`Prefix/Suffix` determines whether the pattern string being checked should appear at the beginning or end of an alias. Alternatively, `Regex` indicates that the pattern string should be treated as a regular expression when matching.

`xconfiguration Zones Zone [1..100] Match [1..5] Pattern Behavior: <Strip/Leave/Replace>`

Determines whether the matched part of the alias should be modified before an LRQ is sent to the indicated zone. If set to `Leave`, the alias will be unmodified. If set to `Strip`, the matching prefix or suffix will removed from the alias. If set to `Replace`, the matching part of the alias will be substituted for the replace text. Note that `Strip` is not a supported option if the pattern type is set to `Regex`.

`xconfiguration Zones Zone [1..100] Match [1..5] Pattern Replace: <S:0, 60>`

The string to be used as a substitution for the part of the alias that matched the pattern.

17.3. Command

The command root command, `xcommand`, is used to execute commands on the Gatekeeper.

To list all `xcommand`s type:

```
xcommand ?
```

To get usage information for a specific command, type:

```
xcommand <command_name> ?
```

17.3.1. AdHocConference

`xCommand AdHocConference <registration>`

Transfers all calls for the specified registration to the configured ad hoc conference.

`registration` specifies the index of the registration whose calls are to be transferred.

17.3.2. AllowListAdd

`xCommand AllowListAdd <allowed_alias>`

Adds an entry to the allow list, used by the registration restriction policy.

`allowed_alias` can either be a specific alias, or use the wildcards **?** (for a single character) and ***** (for a single character or string of characters) to pattern match a group of possible aliases.

17.3.3. AllowListDelete

`xCommand AllowListDelete <index>`

Removes the pattern with the specified index from the allow list.

Allow list entries can be viewed using the command `xconfiguration Gatekeeper Registration AllowList`.

17.3.4. Boot

`xCommand Boot`

Reboots the Gatekeeper. This takes approximately 2 minutes to complete.

17.3.5. CallTransfer

`xCommand CallTransfer <call_index> <leg_index> <alias>`

Attempts to transfer the half of the call identified by the call index and leg to the given alias.

Call and leg indices may be identified using `xstatus calls`.

17.3.6. CheckBandwidth

`xCommand CheckBandwidth <node1> <node2> <bandwidth> <calltype>`

This is a diagnostic function for verifying bandwidth control.

`Node1` `Node2`	The case-sensitive names of the nodes.
`bandwidth`	The required bandwidth.
`calltype`	Must be one of `Traversal`, `Routed` or `Direct`

17.3.7. CredentialAdd

`xCommand CredentialAdd <`*`username`*`> <`*`password`*`>`

Adds the given username and password to the local authentication database.

17.3.8. CredentialDelete

`xCommand CredentialDelete <index>`

Deletes the indexed credential.

17.3.9. DefaultLinksAdd

`xCommand DefaultLinksAdd`

Restores the factory default links for bandwidth control.

17.3.10. DefaultValuesSet

`xCommand DefaultValuesSet Level <level>`

Resets system parameters to default values. Level 1 will reset most parameters. There are currently no level 2 parameters, so setting that level has the same effect as setting level 1. Level 3 resets all level 1 and 2 parameters as well as the following:

- IP address, subnet mask, gateway and interface speed. The default IP address is 192.168.0.100.
- COM port baud rate, speed, data bits, parity, stop bits
- SNMP community name and host address
- system name
- password
- option key
- release key

Note: `DefaltValuesSet` will not add the links with which the system ships from the factory. Use the `DefaultLinksAdd` command to do that. Certificates and policy files are not removed.

17.3.11. DenyListAdd

`xCommand DenyListAdd <`*`denied_alias`*`>`

Add an entry to the deny list. This is used by the registration restriction policy.

`denied_alias` can either be a specific alias, or use the wildcards **?** (for a single character) and ***** (for a single character or string of characters) to pattern match a group of possible aliases.

17.3.12. DenyListDelete

`xCommand DenyListDelete <index>`

Removes the pattern with the specified index from the deny list.

Deny list entries can be viewed using the command `xconfiguration Gatekeeper Registration DenyList`.

17.3.13. Dial

`xCommand Dial <callsrc> <calldst> <bandwidth>`

Places call halves out to the specified source and destination, joining them together.

`callsrc` and `calldst` can be specified using either an alias or IP address.

`Bandwidth` is in kbps.

17.3.14. DisconnectCall

`xCommand DisconnectCall <callid>`

Disconnects the specified call. You can specify the call using either its call index or its serial number, which can be identified using `xstatus call`.

17.3.15. FeedbackRegister

`xCommand FeedbackRegister <ID> <URL> <Expression1>`

Registers for notifications on the event or status change described by the Expression. Notifications are sent in XML format to the specified URL. Up to 15 Expressions may be registered for each of 3 feedback IDs.

The following Expressions are valid:

- `Event`
- `Event/AuthenticationFailure`
- `Event/CallAttempt`
- `Event/Connected`
- `Event/Disconnected`
- `Event/ConnectionFailure`
- `Event/Locate`
- `Event/Registration`
- `Event/ResourceUsage`
- `Event/Unregistration`
- `Event/Bandwidth`
- `Status`
- `Status/Calls`
- `Status/Registrations`
- `History`
- `History/Calls`
- `History/Registrations`

For example: (backslashes are used to indicate continuation lines)

```
xCommand FeedbackRegister ID:1 \
URL:http://10.1.1.1/SystemManagementService.asmx \
Expression:Event/Connected,Status/Calls
```

would notify all call connections and their subsequent changes in status to the specified URL.

17.3.16. FeedbackDeregister

`xCommand FeedbackDeregister <ID>`

Deregisters the specified Feedback Expression.

All registered Feedback Expressions may be removed by issuing the command:

```
xCommand FeedbackDeregister 0
```

17.3.17. FindRegistration

`xCommand FindRegistration <`*`alias`*`>`

Returns information about the registration associated with the specified *`alias`*. The alias must be registered on the Gatekeeper on which the command is issued.

See also `xCommand Locate`.

17.3.18. LinkAdd

`xCommand LinkAdd <`*`linkname`*`> <node1> <node2> <pipe1> <pipe2>`

Adds a new link to the link list with the specified nodes and pipes. The nodes and pipes must already exist on the system.

17.3.19. LinkDelete

`xCommand LinkDelete <index>`

Deletes the link with the specified index.

17.3.20. Locate

`xCommand Locate <alias> <HopCount>`

Runs the Gatekeeper'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 set up before issuing this command.

17.3.21. OptionKeyAdd

`xCommand OptionKeyAdd <key>`

Adds a new option key.

17.3.22. OptionKeyDelete

`xCommand OptionKeyDelete <index>`

Deletes the option key with the specified index.

17.3.23. PipeAdd

`xCommand PipeAdd <`*`name`*`> <totalmode> <total> <percallmode> <percall>`

Adds and configures a new pipe.

17.3.24. PipeDelete

```
xCommand PipeDelete <index>
```

Deletes the pipe with the specified index.

17.3.25. RemoveRegistration

```
xCommand RemoveRegistration <index>
```

Removes the specified registration.

17.3.26. SubZoneAdd

xCommand SubZoneAdd <*name*> <*address*> <prefixlength> <totalmode>

Adds and configures a new subzone.

Parameters include:

name	User assigned label for the subzone.
address	IP address for the sub-zone.
prefixlength	Number of bits which must match for an IP address to be in this subzone.
totalmode	Determines whether bandwidth is controlled for this node. None prevents any calls Limited imposes bandwidth limits Unlimited imposes no bandwidth limits

17.3.27. SubZoneDelete

```
xCommand SubZoneDelete <index>
```

Deletes the subzone with the specified index.

17.3.28. TransformAdd

```
xCommand TransformAdd <pattern> <priority> <type> <behavior> <replace>
```

Adds a new destination alias transform. Parameters are:

pattern	The pattern to match against destination aliases
priority	The priority of the transform
type	The type of matching to apply - options are Prefix, Suffix or Regex
behavior	The action to take for the transform - options are Strip or Replace
replace	The text to be substituted

17.3.29. TransformDelete

`xCommand TransformDelete <index>`

Deletes the transform with the specified index.

Note: a list of all current transforms can be obtained using the command: `xconfiguration gatekeeper transform`.

17.3.30. TraversalZoneAdd

`xCommand TraversalZoneAdd <`*`name`*`> <`*`mode`*`>`

Adds a new traversal zone with the specified name. The mode can be either Assent or H.460.18.

17.3.31. TraversalZoneDelete

`xCommand TraversalZoneDelete <index>`

Removes the traversal zone with the specified index.

17.3.32. ZoneAdd

`xCommand ZoneAdd <`*`name`*`> <`*`Address`*`>`

Adds a new zone with the specified name and address. . This can be either an IP Address or, if DNS is configured, a FQDN. The zone is pre-configured with a link to the default subzone and a pattern match mode of `AlwaysMatch`.

17.3.33. ZoneDelete

`xCommand ZoneDelete <index>`

Removes the zone with the specified index.

17.4. History

The history root command, `xhistory`, is used to display historical data on the Gatekeeper.

To list all xhistory commands type:

```
xhistory ?
```

To list all history data, type:

```
xhistory
```

To show a specific set of history data, type:

```
xhistory <name>
```

17.4.1. calls

`xhistory calls`

Displays history data for up to the last 255 calls handled by the Gatekeeper. Call entries are added to the Call History on call completion. Call histories are listed in reverse chronological order of completion time.

`xhistory calls call <index>`

Displays data for the call with the specified index.

17.4.2. registrations

`xhistory registrations`

Displays history data for up to the last 255 registrations handled by the Gatekeeper. Registration entries are added to the Registration History on unregistration of H.323 entities. Registration histories are listed in reverse chronological order of unregistration time.

`xhistory registrations registration <index>`

Displays data for the registration with the specified index.

17.5. Feedback

The feedback root command, `xfeedback`, is used to control notifications of events and status changes on the Gatekeeper.

A Feedback Expression describes an interesting event or change in status. When a Feedback Expression is registered, a notification will be displayed on the console for each occurrence of the event described by that Expression. Notifications will continue to be displayed for a given event until the Expression is deregistered.

To list all `xfeedback` commands type:

```
xfeedback ?
```

To list all currently active feedback expressions, type:

```
xfeedback list
```

To register a feedback expression, type:

```
xfeedback register <expression>
```

To deregister the feedback expression with index <n>, type:

```
xfeedback deregister <n>
```

To deregister all feedback expressions, type:

```
xfeedback deregister 0
```

17.5.1. Register status

`xfeedback Register Status`

Registers for all status changes.

`xfeedback Register Status/<Calls/Registrations>`

Registers for feedback on changes in the status of either `calls` or `registrations` only.

17.5.2. Register History

`xfeedback Register History`

Registers for feedback on all history.

`xfeedback Register History/<Calls/Registrations>`

Registers for feedback on history of either `calls` or `registrations` only.

17.5.3. Register event

`xfeedback Register Event`

Registers for all available Events.

`xfeedback Register Event/<CallAttempt/Connected/Disconnected/ConnectionFailure/Registration/Unregistration/Bandwidth/ResourceUsage>`

Registers for feedback on the occurrence of the specified Event.

Note: Registering for the `ResourceUsage` event will return the entire `ResourceUsage` structure every time one of the `ResourceUsage` fields changes. `ResourceUsage` fields consist of:

```
Registrations
MaxRegistrations
TraversalCalls
MaxTraversalCalls
TotalTraversalCalls
NonTraversalCalls
MaxNonTraversalCalls
TotalNonTraversalCalls
```

17.6. Other Commands

17.6.1. about

`about`

Returns information about the software version installed on the system.

17.6.2. clear

`clear <eventlog/history>`

Clears the event log or history of all calls and registrations.

17.6.3. eventlog

`eventlog <`*`n`*`/all>`

Displays the event log. The event log contains information about past events which may be useful for diagnostic purposes.

`n`	The number of lines (from end of event log) to display.
`all`	Displays the whole event log.

17.6.4. license

`license`

Returns a list of the third party software licenses incorporated in the product.

`license <index>`

Returns the terms of the license with the specified index.

17.6.5. relkey

`relkey`

Returns the release key with which this software has been installed.

17.6.6. Syslog

`syslog <level> [IPAddress] [IPAddress]`

Enables tracing to the console for the specified IP addresses.

`level`	Specifies the detail at which to trace. Levels are 0-3; 3 gives most logging.
`IPAddress`	Optional parameters which specify up to 10 IP addresses to log information for. If no addresses are specified, activity to all IP addresses will be logged.

Setting `syslog 0` will turn off tracing.

18. Appendix A: Configuring DNS Servers

In the examples below, we set up an SRV record to handle H.323 URIs of the form *user@example.com* These are handled by the system with the fully qualified domain name of *gatekeeper1.example.com* which is listening on port 1719, the default registration port.

It is assumed that an A record already exists for *gatekeeper1.example.com*. If not, you will need to add one.

18.1. Microsoft DNS Server

It is possible to 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 service_data
```

Where:

`domain`	is the domain into which you wish to insert the record
`service_name`	is the name of the service you're adding
`service_data`	is the priority, weight, port and server providing the service as defined by RFC 2782 [3].

For example:

```
dnscmd . /RecordAdd example.com _h323ls._udp SRV 1 0 1719
gatekeeper1.example.com
```

18.2. 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 [6]

18.3. 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.

19. Appendix B: Configuring LDAP Servers

19.1. Microsoft Active Directory

19.1.1. Prerequisites

These comprehensive step-by-step instructions assume that Active Directory is 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.

19.1.2. Adding H.350 objects

1. Create the organizational hierarchy

Open up the Active Directory Users and Computers MMC snap-in. Under your BaseDN right-click and select New Organizational Unit. Create an Organizational unit called h350.

Note: 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 Gatekeeper read access to the BaseDN and therefore limit access to other sections of the directory.

2. Add the H.350 objects

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

commUniqueId: comm1

h323Identityh323-ID: MeetingRoom1

h323IdentitydialedDigits: 626262

h235IdentityEndpointID: meetingroom1

h235IdentityPassword: mypassword
```

Add the `ldif` file to the server using the command:

```
ldifde -i -c DC=X <ldap_base> -f filename.ldf
```

This will add a single H.323 endpoint with an H.323 Id alias of *MeetingRoom1* and an E.164 alias of *626262*. The entry also has H.235 credentials of id *meetingroom1* and password *mypassword* which are used during authentication.

19.1.3. 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.

19.2. OpenLDAP

19.2.1. 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.

19.2.2. Installing the H.350 schemas

The following ITU specification describes 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.

The schemas can be downloaded in `ldif` format from the web interface on the Gatekeeper. To do this, navigate to *Gatekeeper Configuration* > *Files* and click on the links for the LDAP schemas.

Copy the downloaded schemas to the OpenLDAP schema directory:

```
/etc/openldap/schemas/commobject.ldif

/etc/openldap/schemas/h323identity.ldif

/etc/openldap/schemas/h235identity.ldif
```

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
```

The OpenLDAP daemon (slapd) must be restarted for the new schemas to take effect.

19.2.3. Adding H.350 objects

1. Create the organizational hierarchy

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
```

Add the `ldif` file to the server using the command:

```
slapadd -l <ldif_file>
```

This organizational unit will form the BaseDN to which the Gatekeeper will issue searches. In this example the BaseDN will be *ou=h350,dc=my-domain,dc=com*.

Note: 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 Gatekeeper read access to the BaseDN and therefore limit access to other sections of the directory.

2. Add the H.350 objects

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

commUniqueId: comm1

h323Identityh323-ID: MeetingRoom1

h323IdentitydialedDigits: 626262

h235IdentityEndpointID: meetingroom1

h235IdentityPassword: mypassword
```

Add the `ldif` file to the server using the command:

```
slapadd -l <ldif_file>
```

This will add a single H.323 endpoint with an H.323 Id alias of *MeetingRoom1* and an E.164 alias of *626262*. The entry also has H.235 credentials of id *meetingroom1* and password *mypassword* which are used during authentication.

19.2.4. 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 Gatekeeper to verify the server's identity. Once 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.

For more details on configuring OpenLDAP to use TLS consult the OpenLDAP Administrator's Guide.

To configure the Gatekeeper to use TLS on the connection to the LDAP server you must upload the CA's certificate as a trusted CA certificate. To do this, navigate to *Gatekeeper Configuration* > *Files* and upload the certificate.

20. Appendix C: Regular Expression Reference

Regular expressions can be used in conjunction with a number of Gatekeeper features such as alias transformations, zone transformations, CPL policy and ENUM. The Gatekeeper uses POSIX format regular expression syntax.

For an example of regex usage, see *Call screening based on alias* (section 14.5.4).

Following is a list of commonly used special characters in regular expression syntax:

Note: For a detailed description of regular expression syntax see [9].

`.`	Matches any character.
`*`	Matches 0 or more repetitions of the previous match. For example `.*` will match against a sequence of any character.
`+`	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. For example, [a-z] will match against any lower case alphabetical character; [a-zA-Z] will match against any alphabetical character. Note that you can not use special characters within the [] - they will be taken literally. For example [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 be referenced when using replace strings to modify a string that matches a regular expression. For example, 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. For example **`.*@example.(net|com)`** will match against any URI for the domain `example.com` or the domain `example.net`.

21. Appendix D: Technical data

21.1. Technical Specifications

21.1.1. System Capacity

- 2500 registered traversal endpoints
- 100 traversal calls at 384 kbps
- 500 non-traversal calls
- 100 zones

Option keys may restrict the system to a lower capacity than specified above.

21.1.2. Ethernet Interfaces

- 3 x LAN/Ethernet (RJ-45) 10/100 Base-TX (2 disabled)

21.1.3. System Console Port

- 2 x COM ports (front and rear), RS-323 DB-9 connector 2 x USB (disabled)

21.1.4. ITU Standards

- ITU-T H.323 version 5 including Annex O
- ITU-T H.235
- ITU-T H.350

21.1.5. Security Features

- IP Administration passwords
- Management via SSH and HTTPS
- Software upgrade via HTTPS and SCP

21.1.6. System Management

- Configuration via serial connection, Telnet, SSH, HTTP, HTTPS
- Software upgraded via HTTP, HTTPS and SCP

21.1.7. Environmental Data

- Operation temperature: 0C to 35C (32F to 95F)
- Relative humidity: 10% to 90% non-condensing

21.1.8. Physical Dimensions

- Height: 4.35 cm (1.72 inches)
- Width: 42.6 cm (16.8 inches)
- Depth: 22.86 cm (9 inches)
- 1U rack mounted chassis

21.1.9. Hardware MTBF

- Hardware MTBF: 80,479 hours

21.1.10. Power Supply

- 250 Watt
- 90-264V full range @47- 63 Hz

21.1.11. Certification

- LVD 73/23/EC
- EMC 89/366/ECC

21.2. Approvals

This product has been approved by various international approval agencies, among others CSA and Nemko. According to their Follow-Up Inspection Scheme, these agencies also perform production inspections at a regular basis, for all production of TANDBERG's equipment.

The test reports and certificates issued for the product show that the TANDBERG Gatekeeper, Type number TTC2-02, complies with the following standards.

21.2.1. EMC Emission - Radiated Electromagnetic Interference

- EN55022:1994 + A1:1995 + A2:1997 Class A.
- FCC Rules and Regulations 47CFR, Part 2, Part 15.
- CISPR PUB.22 Class A

21.2.2. EMC Immunity

- EN 55024:1998 + A1:2001
- EN 61000-3-2:2000
- EN 61000-3-3:1995 + A1:2001

21.2.3. Electrical Safety

- IEC 60950-1 edition 2001
- EN 60950-1 edition 2001 +A11:2004
- UL 60950-1. 1st Edition
- CSA 60950-1-03

21.2.4. ICSA certification

The TANDBERG Gatekeeper software release 5.x has been certified by the ICSA. Full details of the certification and the lab report are available from:

https://www.icsalabs.com/icsa/docs/html/communities/services/Lab_Reports/Tandberg_cert_02.pdf.

22. Bibliography

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

23. Glossary

Alias

The name an endpoint uses when registering with the Gatekeeper. Other endpoints can then use this name to call it.

ARQ, Admission Request

An endpoint RAS request to make or answer a call.

DNS Zone

A subdivision of the DNS namespace. *example.com* is a DNS zone.

E.164

An ITU standard for structured telephone numbers. Each telephone number consists of a country code, area code and subscriber number. For example, TANDBERG's European Headquarters' phone number is +47 67 125125, corresponding to a country code of 47 for Norway, area code of 67 for Lysaker and finally 125125 to determine which phone line in Lysaker.

External Manager

The remote system that is used to manage endpoints and network infrastructure. The TANDBERG Management Suite (TMS) is an example of an external manager.

Gatekeeper Zone

A collection of all the endpoints, gateways and MCUs managed by a single gatekeeper.

LRQ, Location Request

A RAS query between Gatekeepers or Border Controllers to determine the location of an endpoint.

RAS, Registration, Admission and Status Protocol

A protocol used between endpoints and a Gatekeeper to register and place calls.

Traversal call

An H.323 call which uses a Border Controller. The Border Controller cooperates with the endpoint or TANDBERG Gatekeeper to allow communication through a firewall. All signaling and media is routed through the Border Controller.

Zone

See *DNS Zone* and *Gatekeeper Zone*.

24. Index