VMware Smart Assurance IP Manager User Guide

VMware Smart Assurance 10.0.0

vmware®

You can find the most up-to-date technical documentation on the VMware website at:

https://docs.vmware.com/

If you have comments about this documentation, submit your feedback to

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2019 VMware, Inc. All rights reserved. Copyright and trademark information.

Contents

1 Preface 6
- Purpose 6
- Audience 6

2 VMware Smart Assurance IP Manager installation directory 7

3 VMware Smart Assurance IP Manager products 8

4 IP Manager documentation 9

5 Related documentation 10

6 Conventions used in this document 11
- Typographical conventions 11

7 Where to get help 12

8 Your comments 13

9 Global Manager and Global Console Tasks 14
- Opening the Global Console 14
- Opening the Polling and Thresholds Console 16
- Customizing the Global Console 16
- Opening the Domain Manager Administration Console 16
- Network notifications 18

10 Configuration 22
- Configuration overview 22
- Configuring the IP Manager 23
- Configuring the Global Manager for communication with underlying domains 40
- Configuring the Adapter Platform 44
- Configuring IP to enable light discovery for Satellite Domain Manager 52
- Example of configuring interface-matching filters 55
- Optimizing the discovery of ERX devices 58
- Enabling the creation of a physical connection for a shared MAC address 59
- Running IP server as non root on UNIX and Linux systems 60
- Upgrade and migration impacts 62

11 Discovery 64

Discovery overview 64

Preparing for discovery 65

Discovery options 67

Initiating discovery 73

Preparing for manual discovery 76

Preparing for autodiscovery 77

Scheduling automatic discovery 81

Starting discovery 83

Creating discovery filters 85

OID Info Tool 89

Seed file related tasks 93

Discovery and short discovery of IPSec tunnels 94

Customizing discovery 95

12 Polling And Threshold 98

Overview of Polling and Thresholds Console 98

Opening the Polling and Thresholds Console 99

Working with groups and settings 101

Performing tasks in the Polling tab 106

Performing tasks in the Thresholds tab 108

External Polling Support for FibreChannelSwitchPort 114

Policy Settings to control the managed state of interfaces 114

Performing tasks in the Device Access tab 126

Performing tasks in the Tagging tab 128

13 Audit Logging 136

Audit Logging format 136

Installing the Audit Log Adapter 138

Configuring audit logging 139

Starting the Audit Log Adapter 139

These two files are located in the *local/logs* directory relative to the adapter. So, if the adapter is installed in a separate base directory than the server it is auditing, the adapter will read the raw audit log file from the server *local/logs* directory and write the final audit log file to the adapter *local/logs* directory.Configuring audit logging 141

Starting the Audit Log Adapter 141

Domain Manager Administrator Console user actions 143

14 Name Resolution 147

Overview 147

Configuring automatic name resolution 148

Configuring exceptions to automatic name resolution 150

Configuring seed-name name resolution 150
Configuring exceptions to seed-name name resolution 152
Three name resolution — Related tasks 153
Local exceptions to automatic and seed-file name resolution 153

15 Working with topology 155
Adding read community strings 155
Deleting read community strings 156
Specifying additional read community strings 156
Observing the management state of an object 156
Setting the management state of an object 156
Managing or unmanaging filesystems 158
Topology Change Analysis Tool (TCAT) 158

16 Network Attached Storage 162
Overview 162
Installing and starting the HTTPS Adapter service 163
Configuring Celerra devices for NAS Extension functionality 164
Configuring IP Availability Manager for NAS autodiscovery 165
Viewing NAS notifications 166
Viewing NAS service topology in maps 167
Opening NAS containment 168

17 Redundancy Group Creation 169
Creation of card redundancy groups 169
Creation of system redundancy groups 169
Creation of network adapter redundancy groups 170

Preface

1

As part of an effort to improve its product lines, VMware Smart Assurance periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your VMware Smart Assurance technical support professional if a product does not function properly or does not function as described in this document.

Note This document was accurate at publication time. Go to VMware Smart Assurance Online Support docs.vmware.comto ensure that you are using the latest version of this document.

This chapter includes the following topics:

- Purpose
- Audience

Purpose

This document is part of the VMware Smart Assurance IP Manager documentation set. It provides the following information:

- Viewing and configuring the Global Manager and Global Console.
- Tasks involved in configuring IP Manager.
- Tasks involved in discovering devices using IP Manager.
- Tasks involved in configuring groups and settings.
- Working with Audit log adapter
- Working with Topology
- Installing and Configuring NAS Extension on IP Manager to discover NAS devices.

Audience

This document is intended for IT managers and system administrators.

VMware Smart Assurance IP Manager installation directory

2

In this document, the term BASEDIR represents the location where VMware Smart Assurance software is installed.

- For UNIX, this location is: /opt/InCharge/<productsuite>.

  On UNIX operating systems, IP Manager is, by default, installed to: /opt/InCharge/IP/smarts. This location is referred to as BASEDIR/smarts.

  Optionally, you can specify the root of BASEDIR to be something different, but you cannot change the <productsuite> location under the root directory.

  The VMware Smart Assurance System Administration Guide provides more information about the directory structure of VMware Smart Assurance software.

VMware Smart Assurance IP Manager products

3

The VMware Smart Assurance IP Manager includes the following products:

- VMware Smart Assurance IP Availability Manager
- VMware Smart Assurance IP Performance Manager
- VMware Smart Assurance IP Availability Manager Extension for NAS

IP Manager documentation

4

The following documents are relevant to users of the IP Manager product. Electronic versions of the updated manuals are available at VMware Smart Assurance Online Support: docs.vmware.com

- VMware Smart Assurance IP Manager Release Notes
- *VMware Smart Assurance Open Source License and Copyright Information*
- *VMware Smart Assurance Installation Guide for SAM, IP, ESM, MPLS, and NPM Managers*
- *VMware Smart Assurance IP Manager Deployment Guide*
- *VMware Smart Assurance IP Manager Concepts Guide*
- *VMware Smart Assurance IP Manager User Guide*
- VMware Smart Assurance IP Manager Reference Guide
- VMware Smart Assurance Topology Split Manager User Guide
- VMware Smart Assurance IP Manager Troubleshooting Guide
- *VMware Important Notice*
- *VMware Smart Assurance IP Manager Certification Matrix*

Related documentation

5

The following VMware Smart Assurance publications provide additional information:

- *VMware Smart Assurance System Administration Guide*
- *VMware Smart Assurance Common Information Model (ICIM) 1.11 Reference for Non-Service Assurance Manager Products*
- *VMware Smart Assurance Common Information Model (ICIM) Reference for Service Assurance Manager*
- Smarts Foundation EMC Data Access API (EDAA) Programmer Guide
- *VMware Smart Assurance ASL Reference Guide*
- *VMware Smart Assurance Perl Reference Guide*
- *VMware Smart Assurance MODEL Reference Guide*
- *VMware Smart Assurance Dynamic Modeling Tutorial*

Conventions used in this document

6

VMware Smart Assurance uses the following conventions for special notices:

Note NOTICE is used to address practices not related to personal injury.

A note presents information that is important, but not hazard-related.

An important notice contains information essential to software or hardware operation.

Typographical conventions

VMware Smart Assurance uses the following type style conventions in this document:

Bold	Use for names of interface elements
Italic	Use for full titles of publications referenced in text
`Monospace`	Use for: ■ System output, such as an error message or script ■ System code ■ Pathnames, filenames, prompts, and syntax ■ Commands and options
`Monospace italic`	Use for variables.
`Monospace bold`	Use for user input.
[ ]	Square brackets enclose optional values
|	Vertical bar indicates alternate selections — the bar means “or”
{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate nonessential information omitted from the example

Where to get help

7

VMware Smart Assurance support, product, and licensing information can be obtained as follows:

Product information

For documentation, release notes, software updates, or information about VMware Smart Assurance products, go to VMware Online Support at:

docs.vmware.com

Technical support

Go to VMware Online Support and click Service Center. You will see several options for contacting VMware Technical Support. Note that to open a service request, you must have a valid support agreement. Contact your VMware sales representative for details about obtaining a valid support agreement or with questions about your account.

Your comments

8

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Provide your feedback on the below link:

docs.vmware.com.

Global Manager and Global Console Tasks

9

This chapter includes the following topics:

- Opening the Global Console
- Opening the Polling and Thresholds Console
- Customizing the Global Console
- Opening the Domain Manager Administration Console
- Network notifications

Opening the Global Console

To open the Global Console:

1 Start the **Global Console**.

- On a UNIX system, go to the BASEDIR/smarts/bin directory in the Service Assurance Manager (Global Manager) installation area and type:

  sm_gui

  Press **Enter**. The **Attach Manager** dialog box opens as shown in Attach Manager dialog box.

  Figure 9-1. Attach Manager dialog box

2 In the dialog box:

3 Ensure that the VMware Smart Assurance Broker for your deployment appears in the **Broker** text box.

4 Click the **Manager** list box or the **Browse** button to display a list of active (running) Managers, and from that list select the Domain Manager to which you want to connect.

5 Type your login username and password.

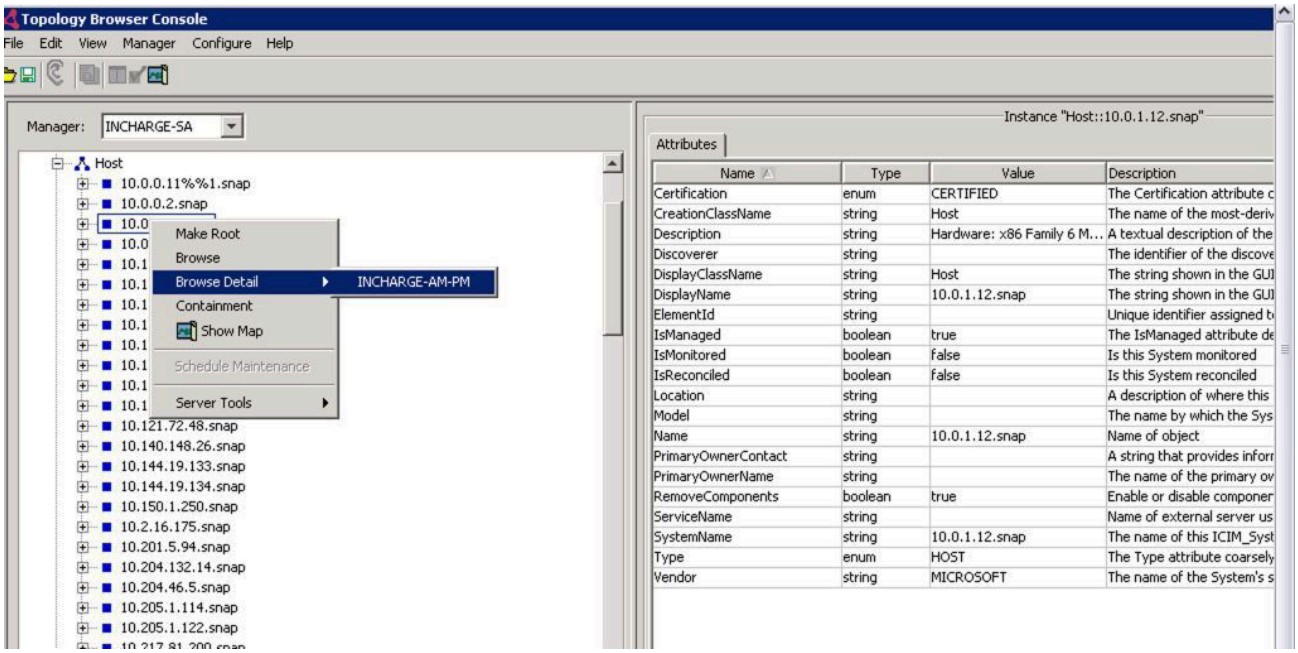
6 Click **OK**.

A console view opens.

What console view opens at this point depends on whether you selected a Domain Manager or a Global Manager application.

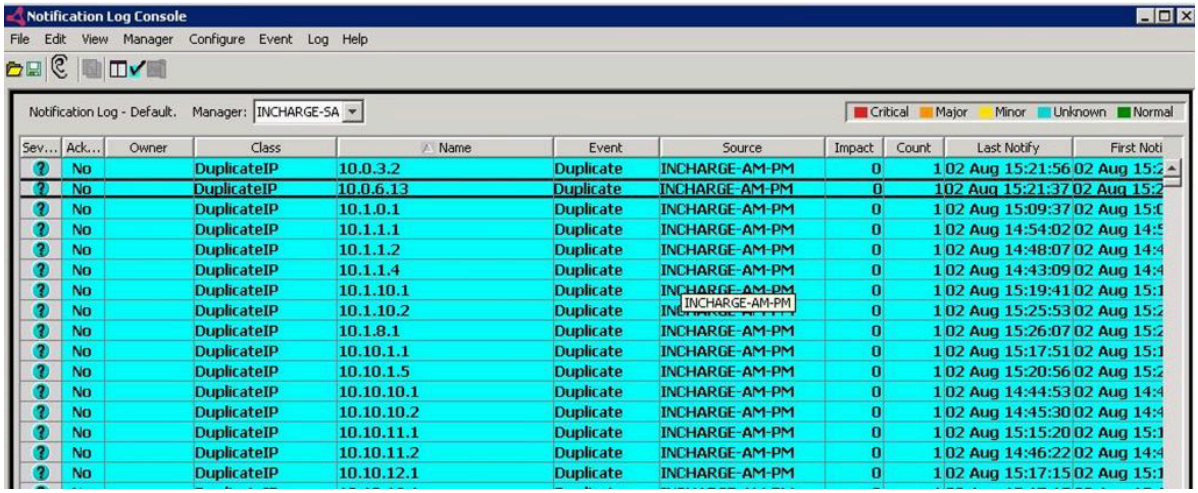
A Topology Browser Console view of the Global Console will open by default, an example of which is shown in *Figure 2 on page 17*.

Figure 9-2. Topology Browser Console



If you selected a Global Manager, a Notification Log Console view of the Global Console will open by default, an example of which is shown in Notification Log Console. In the example display, the Notification Log Console is attached to a Global Manager application named INCHARGE-SA.

Figure 9-3. Notification Log Console



The VMware Smart Assurance Service Assurance Manager Operator Guide provides detailed instructions on using the Global Console.

Opening the Polling and Thresholds Console

You can open a Polling and Thresholds Console view from any Domain Manager Administration Console by selecting the **Configure > Polling and Thresholds** menu option.

Customizing the Global Console

As explained in the VMware Smart Assurance Service Assurance Manager Operator Guide, the -D option for the **sm_gui** command enables you to adjust specific Java properties that affect the Global Console. For example, you can control the display of text for toolbar buttons, maps, and the font that displays throughout the console. The -D option also enables you to override certain default Global Console behavior.

The VMware Smart Assurance Service Assurance Manager Operator Guide provides more information on the -D option for each property or value pair.

Opening the Domain Manager Administration Console

You can open a Domain Manager Administration Console view from any Global Console that is attached to a Domain Manager or a Domain Manager component server, such as the Topology Browser Console in *Figure 2 on page 17*, by selecting the **Configure > Domain Manager Administration Console** menu option.

To open the Domain Manager Administration Console:

1 Start the **Global Console**:

- On a UNIX system, go to the BASEDIR/smarts/bin directory in the Service Assurance Manager (Global Manager) installation area and type:

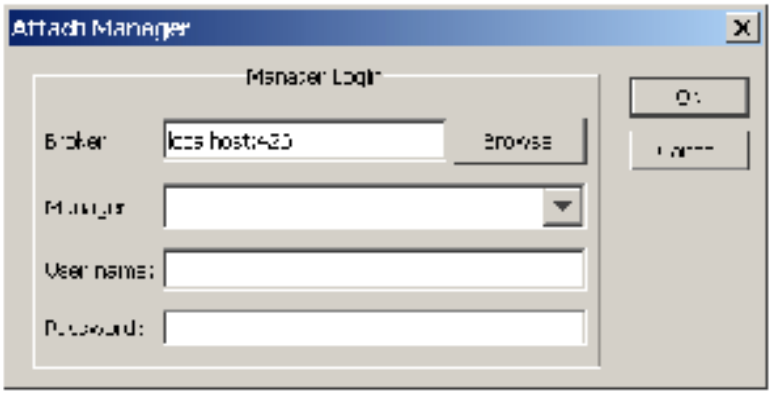
  sm_gui

  Note To view more sm_gui options, type *sm_gui --help*.

2 Press **Enter**.

The **Attach Manager** dialog box opens as shown in Attach Manager dialog box.

Figure 9-4. Attach Manager dialog box



3 In the dialog box:

4 Ensure that the VMware Smart Assurance Broker for your deployment appears in the **Broker** text box.

5 Click the **Manager** list box or the **Browse** button to display a list of active (running) Managers, and from that list select the Domain Manager (for example, INCHARGE-AM) in your deployment as the Manager to which you want to connect.

6 Type your login username and password.

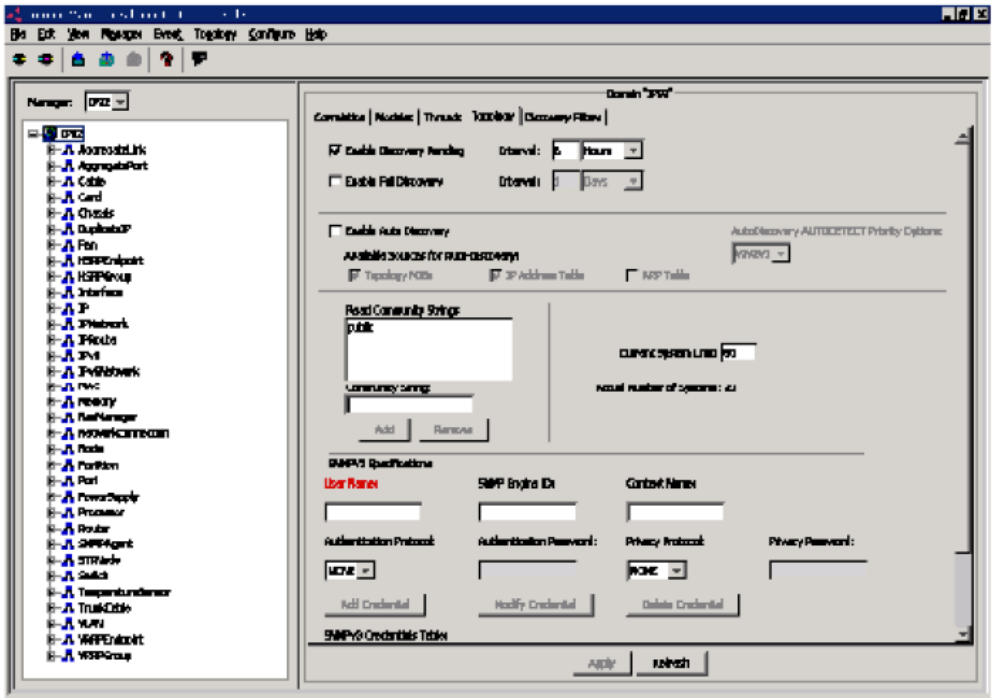
7 Click **OK**.

The **Topology Browser Console** opens. If you selected a Global Manager, a **Notification Log Console** view of the Global Console will open by default.

8 On the **Topology Browser Console**, select **Configure** > **Domain Manager Administration Console**. The **Domain Manager Administration Console** opens, an example of which is shown in Domain Manager Administration Console.

In the example display, the **Domain Manager Administration Console** is attached to an IP Availability Manager application named IP92.

Figure 9-5. Domain Manager Administration Console



The Domain Manager Administration Console is the primary tool for configuring discovery, initiating discovery, and managing topology for a Domain Manager.

The VMware Smart Assurance Service Assurance Manager Operator Guide provides detailed descriptions of the administration capabilities that are available through the Domain Manager Administration Console.

Network notifications

IP Availability Manager passes the results of its analysis in the form of notifications to the Global Manager. The Global Manager, in turn, passes the notifications to the Global Console, where they are presented in a tabular form in the Notification Log Console.

Each notification consists of a set of attributes that describe properties of the notification, such as Class, Name, Event, Source, and Impact. The value in the Name attribute is the display name of the notification.

Chapter 4, Network Topology Naming Conventions, in the *VMware Smart Assurance IP Manager Reference Guide* describes the naming formats used to create network object names.

The numeric value for the Impact attribute indicates the effect of this notification on other objects in the managed network; the larger the numeric value, the larger the effect.

The severity of a notification is reflected by its color, where red indicates the highest severity level.

Viewing network notifications

You can view the notifications through the Global Console in two basic ways:

- As tabular entries in a Notification Log Console view

  Included with the Notification Log Console view (or any other view where you can see notifications) is a Notification Properties dialog box for viewing detailed information about an individual notification.

- As color-coded severity bar icons in a Map Console view

  In a map, a color-coded severity bar icon appears underneath nodes (graphical representations of topology objects) that are affected by abnormal conditions denoted in notifications.

Opening a Notification Properties dialog box

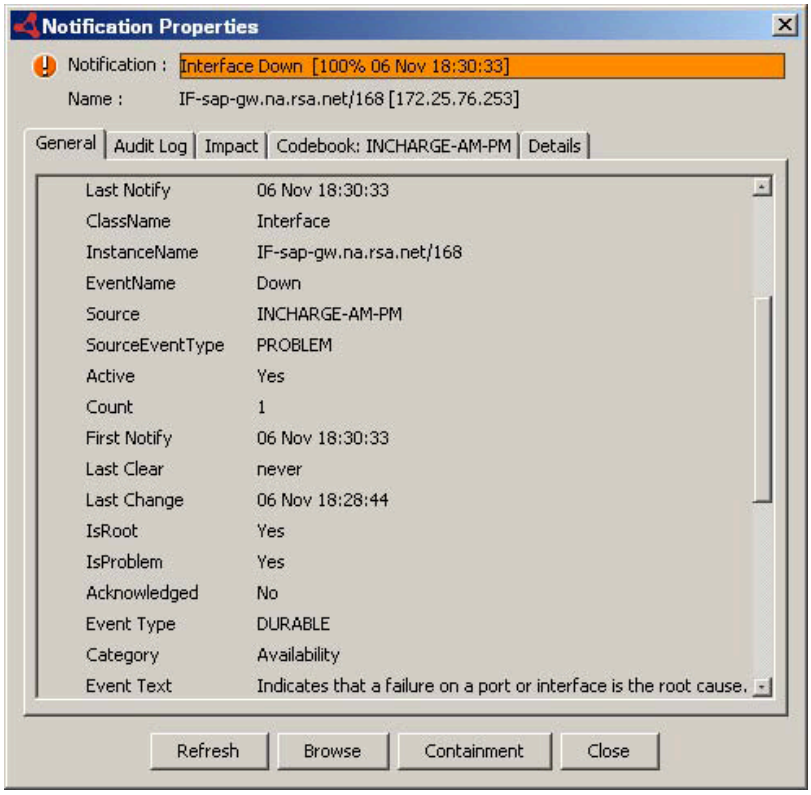
To obtain detailed information about an individual network notification, you can use any of the following common methods to open the Notification Properties dialog box:

- Double-click a network notification in the **Notification Log Console**.
- Select a network notification in the **Notification Log Console**, and then click the **Show Notification Properties** toolbar button.
- Right-click a selected network notification, and then select **Properties** in the pop-up menu.
- Double-click a map icon (in the Map Console) affected by active events.

  Depending on the type of event and information available for display, the Notification Properties dialog box displays up to seven tabs. The tabs are: General, Audit Log, Codebook, Impact, Details, Caused By, and Aggregates.

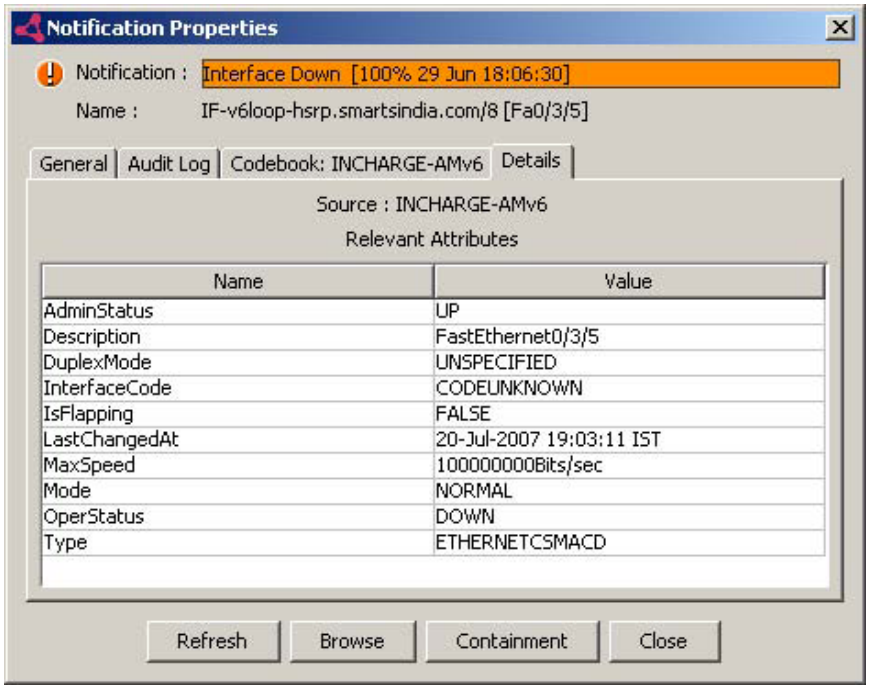
  Notification Properties dialog box (IPv4 example) is an example of a Notification Properties dialog box showing information for an Interface Down problem.

Figure 9-6. Notification Properties dialog box (IPv4 example)



Notification Properties dialog box (IPv6 example) is an example of a Notification Properties dialog box showing information for an Interface Down notification.

Figure 9-7. Notification Properties dialog box (IPv6 example)



In Notification Properties dialog box (IPv4 example) and Notification Properties dialog box (IPv6 example) attribute IsRoot set to Yes indicates that Interface Down is deemed a root-cause problem by the Global Manager, and attribute IsProblem set to Yes indicates that Interface Down is deemed a root-cause problem by IP Availability Manager.

Configuration

10

This chapter includes the following topics:

- Configuration overview
- Configuring the IP Manager
- Configuring the Global Manager for communication with underlying domains
- Configuring the Adapter Platform
- Configuring IP to enable light discovery for Satellite Domain Manager
- Example of configuring interface-matching filters
- Optimizing the discovery of ERX devices
- Enabling the creation of a physical connection for a shared MAC address
- Running IP server as non root on UNIX and Linux systems
- Upgrade and migration impacts

Configuration overview

The following sections highlight the configuration tasks that are associated with the setup of the application components in an IP Manager deployment. The following are the tasks involved:

- Configuring the IP Manager
- Configuring the Global Manager for communication with underlying domains
- *"Configuring the Adapter Platform" on page 49*
- *"Configuring IP to enable light discovery for Satellite Domain Manager" on page 57*

  Some configuration tasks are performed before the applications are started, such as the editing of configuration files, and some are performed through the Global Console when the applications are up and running.

Configuring the IP Manager

Configuring the IP Manager consists of the following tasks:

- Configure the IP Manager with the SM_IP_VERSIONS environment variable for name resolution and server-to-server communication.
- Configure the IP Manager to discover, monitor, and analyze the network devices.
- Enable light discovery for the satellite Domain Managers (optional).
- Configure ICMP and SNMP polling for IP Availability Manager and IP Performance Manager (optional).
- Configure ICMP and SNMP thresholds for the IP Availability Manager and IP Performance Manager (optional).
- Create CLI device-access objects for the satellite Domain Managers (optional)
- Create IP tag filters to enable IP Availability Manager and IP Performance Manager to model overlapping IP addresses (optional).
- Configure policies to manage discovered objects, to determine which objects are to be monitored by the IP Managers (optional).

  "Policy Settings to control the managed state of interfaces" on page 118, provides information for performing this task.

Methods of modifying user configuration parameters

There are two methods to modify user configuration parameters:

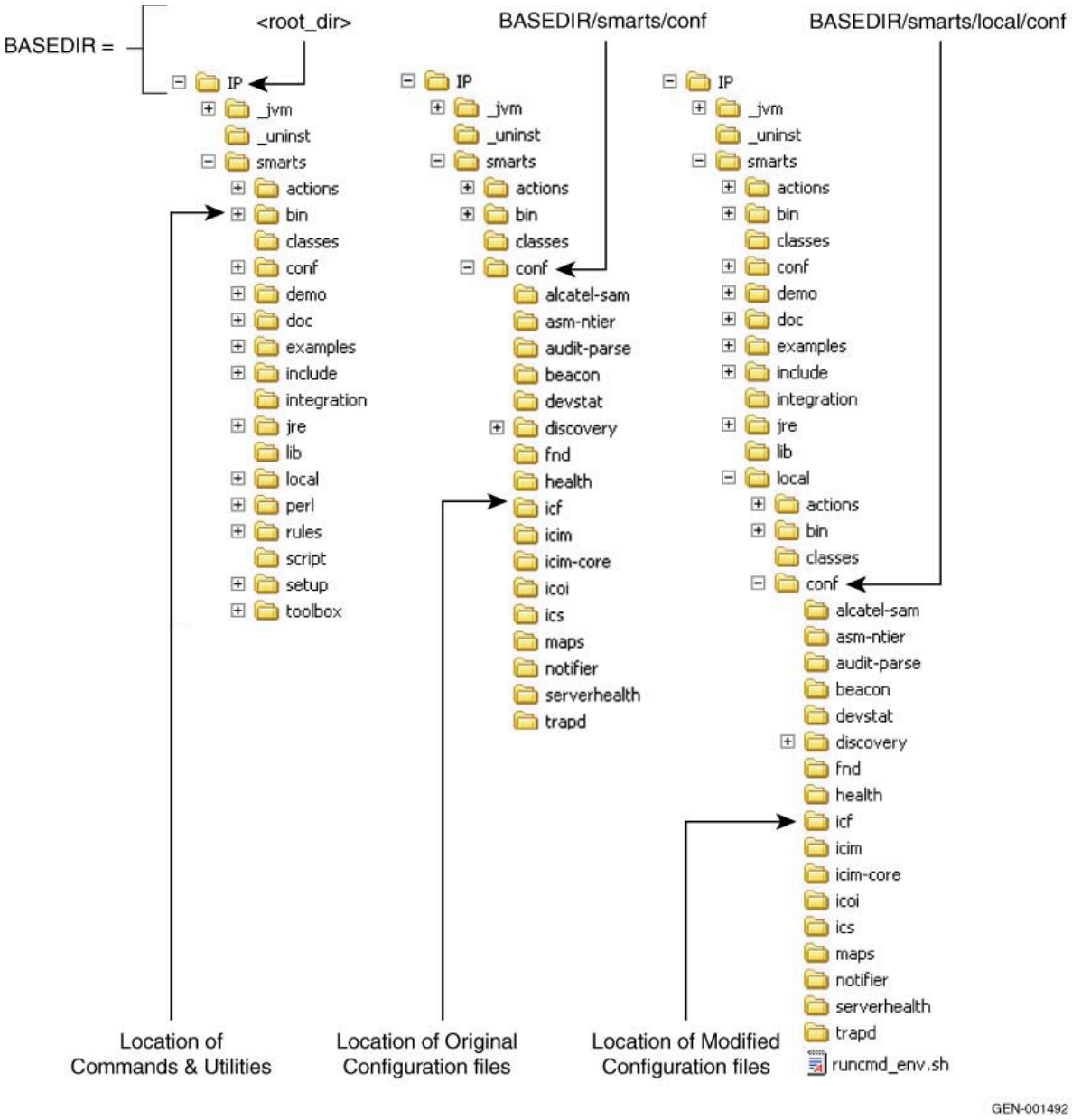
1. Edit the user configuration files and modify configuration parameters.
2. Issue commands to modify configuration parameters when the IP Manager is running.

   Command-modified parameters are not persistent, which means that the modifications will be forgotten when the IP Manager is shut down and restarted. Every time that the IP Manager starts up, it sets its configurations in accordance to the parameter values in the user (and other) configuration files.

Editing the user configuration files and modify configuration parameters

VMware Smart Assurance provides the **sm_edit** utility to ensure that modified files are always saved to the appropriate local area, and that original copies of the files remain unchanged. Preview of how the sm_edit utility works previews the operation of the **sm_edit** utility.

Figure 10-1. Preview of how the sm_edit utility works



All VMware Smart Assurance product suites use the same basic installation directory structure that is shown in Figure 8 on page 27.

To invoke the **sm_edit** utility,

1 Navigate to BASEDIR/smarts/bin directory.

2 In the command line, specify the path and the name of the file, relative to the BASEDIR/smarts directory, that you want to edit. For example,

```
sm_edit conf/discovery/discovery.conf
```

3 Modify and save the discovery.conf file. The **sm_edit** utility saves the modified version of the file to the BASEDIR/smarts/local/conf/discovery directory.

   You can use the **sm_edit** utility to edit any text file, not just a configuration file, in the BASEDIR/smarts or BASEDIR/smarts/local directory. Because sm_edit assumes a starting point of BASEDIR/smarts, the text-file path that you specify begins with the directory name (conf, rules, script, and so on) under the BASEDIR/smarts directory.

   Note Original versions of files may be changed, updated, or moved as part of an VMware Smart Assurance software upgrade. The new patch files from the upgrade are written to BASEDIR/smarts/local directory and the unchanged patch files are removed. All files that contain custom modifications are moved to BASEDIR/smarts/.local.bkp.<version>

   The VMware Smart Assurance System Administration Guide provides additional information about the **sm_edit** utility.

Issuing commands to modify configuration parameters when the IP Manager is running

You can issue any of the following commands from the BASEDIR/smarts/bin directory to change configuration parameters for the IP Manager:

- dmctl put
- dmctl invoke
- dmctl get
- sm_tpmgr

  The **dmctl** utility is described in the HTML pages that are located in the BASEDIR/smarts/doc/html/usage directory of any Domain Manager installation area. VMware Smart Assurance IP Manager Reference Guide, Chapter 8 - IPv6 Address Conventions, provides additional syntax information. You can type "?" in dmctl to see a list of all available commands.

dmctl put command and example

In general, you can use the dmctl put command to change the value of any basic-type attribute (string, boolean, integer, float, and so on) of any VMware Smart Assurance object. Upon issuing a dmctl command, you might be prompted for a username and password. Respond with your user account. For example, to specify the default administrative account, enter username *admin* and password *changeme*.

For IP Manager, you use the dmctl put command to override parameter settings in the discovery.conf file or the name-resolver.conf file. These parameters are defined as attributes of the ICF-TopologyManager object, which is an instance of the ICF_TopologyManager class.

The syntax for the dmctl put command is:

```
dmctl -s <IP Manager instance name> put

      ICF_TopologyManager::ICF-TopologyManager::<configuration property> <value>
```

For example:

```
dmctl -s INCHARGE-AM put ICF_TopologyManager::ICF-TopologyManager::numberProbeThreads 12
```

dmctl get command and example

You can use the dmctl get command to verify the value of the attribute that has been set.

The syntax for the dmctl get command is:

```
dmctl -s <IP Manager instance name> get

ICF_TopologyManager::ICF-TopologyManager::<configuration parameter name>
```

For example:

```
dmctl -s INCHARGE-AM get ICF_TopologyManager::ICF-TopologyManager:: HSRPEnabled
```

dmctl invoke command and example

You use the dmctl invoke command to override parameter settings in the tpmgr-param.conf file. These parameters are entries for a table that is defined inside the ICF-TopologyManager object.

The syntax for the dmctl invoke command is:

```
dmctl -s <IP Manager instance name> invoke

      ICF_TopologyManager::ICF-TopologyManager insertParameter

   <configuration parameter name> <value>
```

For example:

```
dmctl -s INCHARGE-AM invoke ICF_TopologyManager::ICF-TopologyManager insertParameter
DisableMPLSDiscovery FALSE
```

sm_tpmgr command and example

You use the sm_tpmgr command to reload the parameters in the following files:

- discovery.conf
- tpmgr-param.conf
- name-resolver.conf file, or

- Any other IP Manager user configuration files

Note A discovery cycle should not be in progress when you issue this command.

After you use the sm_edit utility to make a parameter change in the discovery.conf, tpmgr-param.conf, or name-resolver.conf file, you can use the sm_tpmgr command to make that change known to the target IP Manager instance without the need to restart the target server. The syntax for the sm_tpmgr command is:

```
sm_tpmgr -b <host>:<port> -s <IP Manager instance name> --load-conf=<configuration file name>
```

For example:

```
sm_tpmgr -b localhost:426 -s INCHARGE-AM --load-conf=user-defined-connections.conf
```

You have to reverse your steps because IFTypePattern-.1.3.6.1.4.1.9.1.222 ~6 is now in the topology. Removing IFTypePattern-.1.3.6.1.4.1.9.1.222 ~6 from tpmgr-param.conf and then reloading it using sm_tpmgr does not remove it from the topology. You must actively remove it.

Try IFTypePattern-.1.3.6.1.4.1.9.1.222 6. (remove ~).

Then reload using sm_tpmgr.

Note Setting it to blank IFTypePattern-.1.3.6.1.4.1.9.1.222 <blank>, would not remove it either. Your modification has to be reversed.

Temporary versus permanent change

The use of the **dmctl** utility to override a parameter value in a configuration file is temporary in the following sense: When the Domain Manager is restarted, it reads the original parameter value in the configuration file. To make the change permanent, use the **sm_edit** utility to change the parameter value in the configuration file.

Limitations

Although you can use the **dmctl** utility to override parameter values in most configuration files, your change might not take effect because a Domain Manager reads some parameters *only* at startup.

For parameter settings that cannot be overwritten using the dmctl or the sm_tpmgr commands, use the **sm_edit** utility to modify the settings in the appropriate configuration files and then restart the IP Manager. Parameter setting changes requiring a restart of the IP Manager identifies the parameters for which a restart is required.

Table 10-1. Parameter setting changes requiring a restart of the IP Manager

Configuration file	Parameter
discovery.conf	DiscoveryAddrPref
	HostResourcesEnabled
	HSRPEnabled

Table 10-1. Parameter setting changes requiring a restart of the IP Manager (Continued)

Configuration file	Parameter
	VRRPEnabled
	VirtualRouterEnabled
tpmgr-param.conf	DisableOverlappingIP
	OITrapAdapterPort
	UserDefinedConnectionsOverwrite
name-resolver.conf	none
partition.conf1	All
user-defined-connections.conf1	All
trapd.conf	PORT
trap-notify.conf	none
runcmd_env.sh 1	All

1 Parameters in this file cannot be overridden by using the dmctl utility; use the sm_edit utility to modify this file.

Enabling connections between devices that are Cisco Discovery Protocol (CDP) neighbors

To enable the IP Manager to bypass the port bridges relationship check while creating port-to-port or port-to-interface connections between two devices that are Cisco Discovery Protocol (CDP) neighbors across LAN segments, add the following parameter and value to the discovery.conf file:

```
CiscoOnlyNetwork TRUE
```

By default, CiscoOnlyNetwork is set to FALSE.

Also, by default, the IP Manager checks a port's "Bridges" relationship for a relationship to a peer port's MAC or a peer interface's MAC. If no MAC is found, which is often the case in a shared media like Ethernet or token ring, the IP Manager assumes that an intermediate device exists between the port and its peer, and responds by not creating a connection between the port and its peer.

In contrast, for CiscoOnlyNetwork = TRUE, the IP Availability Manager:

- Assumes that all networking devices are Cisco and are running CDP, and that no intermediate device exists between the port and its peer.
- Bypasses the port Bridges relationship check.
- Creates a connection between the port and its peer.

Fine tuning ICMP and SNMP discovery related parameters

The discovery.conf file contains several settings that you can edit to improve performance during topology discovery. However, changing these values beyond the recommended values can adversely affect the discovery process. These settings include:

- defaultICMPAutoRetries
- defaultICMPAutoTimeout
- defaultSNMPAutoRetries
- defaultSNMPAutoTimeout
- defaultRetries
- defaultTimeout

  The defaultICMPAutoRetries, defaultICMPAutoTimeout, defaultSNMPAutoRetries, and defaultSNMPAutoTimeout settings apply to discovery. These settings do not affect the operation of the ICMP or SNMP poller that polls objects to perform analysis.

  When a timeout occurs because of packet loss in the network, increase the number of retries. If the timeout occurs because the network is slow or the system is busy, increase the value of the timeout.

ICMP discovery settings

Before sending an SNMP request, the discovery process sends an ICMP poll to the system to ensure it is reachable. The defaultICMPAutoRetries and defaultICMPAutoTimeout control the ICMP poll that is used during the initial stage of discovery.

Retries is the number of times an ICMP poll is sent to a system after the initial attempt fails. The default value is 3. Timeout is the amount of time that should elapse before considering that the previous ICMP poll has failed or timed out. The default value is 500 milliseconds.

The value of the timeout is constant, meaning that the value stays the same between retries. In contrast, the value of the timeout for the SNMP poller doubles after each retry.

SNMP discovery settings

The defaultSNMPAutoRetries and defaultSNMPAutoTimeout settings control SNMP polling during the first phase of the discovery process.

The defaultSNMPAutoRetries is the number of times that the SNMP poller should try to reach a system after the initial attempt fails. The default value is 3. The defaultSNMPAutoTimeout is the amount of time that the SNMP poller should wait before it considers an SNMP request to have failed or timed out. The default value is 1000 milliseconds. The defaultRetries and defaultTimeout settings control the SNMP poller during the third phase of the discovery process, when the systems are fully probed and the topology is created.

The defaultRetries is the number of times that the SNMP poller should try to reach a system after the initial attempt fails. The default value is 5. The defaultTimeout is the amount of time that the SNMP poller should wait before it considers an SNMP request to have failed or timed out. The default value is 1000 milliseconds.

VMware recommends that you adjust the timeout and retries settings as follows:

- If random systems are experiencing excessive timeouts, a slow system response time is the most probable cause. In this case, increase the time-out value by 50 percent and decrease the number of retries by 1.
- If several systems within the same subnet are experiencing excessive timeouts, a high-packet loss rate is the most probable cause. In this case, increase the number of retries by 1 and decrease the time-out value by 50 percent.

  Do not increase the value of the timeout beyond 8,000 milliseconds. The timeout doubles with each retry. Therefore, an initial value of 8,000 milliseconds with the default number of retries (5) results in a cumulative timeout interval of 504,000 milliseconds (more than 8 minutes) during the discovery process. In addition, because increasing the number of retries by 1 effectively doubles the total timeout, do not increase both the timeout and the number of retries.

Controlling the IP version for name resolution

If a hostname includes an explicit IP version suffix (:v6v4, :v4v6, :v6, or :v4), the local name resolution service, such as a Domain Name System server, resolves the hostname to an address of that protocol. Here are some examples:

- frame.vmware.com:v6v4

  Resolves to an IPv6 address, or if that resolution fails, to an IPv4 address.
- frame.vmware.com:v4v6

  Resolves to an IPv4 address, or if that resolution fails, to an IPv6 address.
- frame.vmware.com:v6

  Resolves to an IPv6 address.
- frame.vmware.com:v4

  Resolves to an IPv4 address.

  For a hostname that does not include an explicit IP version suffix, the SM_IP_VERSIONS environment variable controls how the local name resolution service resolves the hostname to a corresponding IP address. By default, the SM_IP_VERSIONS environment variable set to v6v4.

  Chapter 8, IPv6 Address Conventions, in the VMware Smart Assurance IP Manager Reference Guide provides additional information about name resolution.

SM_IP_VERSIONS environment variable

Valid values for the SM_IP_VERSIONS environment variable are:

- v6v4 (default)

  Hostname is resolved to an IPv6 address. If that resolution fails, the local name resolution service tries to resolve the hostname to an IPv4 address.

- v4v6

  Hostname is resolved to an IPv4 address. If that resolution fails, the local name resolution service tries to resolve the hostname to an IPv6 address.

- v6

  Hostname is resolved to an IPv6 address.

- v4

  Hostname is resolved to an IPv4 address.

  Values v6v4, v4v6, v6, and v4 are not case-sensitive.

Setting SM_IP_VERSIONS

The SM_IP_VERSIONS environment variable is set to v6v4 through the following definition in the runcmd_env.sh file:

```
SM_IP_VERSIONS=v6v4
```

The runcmd_env.sh file in the IP Manager installation area will automatically set the SM_IP_VERSIONS environment variable, along with any other environment variables in the file, for each application or utility that is started from the installation area.

You should set the SM_IP_VERSIONS environment variable to the IP version that is predominant for your managed network. For example, use the v4v6 value if most systems in your managed network are identified by IPv4 addresses, and some are identified by IPv6 addresses.

To set the SM_IP_VERSIONS environment variable to a value other than v6v4:

1. Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the runcmd_env.sh file:

   ```
   sm_edit conf/runcmd_env.sh
   ```

2. Find the following definition in the file:

   ```
   SM_IP_VERSIONS=v6v4
   ```

3 Change the value of SM_IP_VERSIONS to v4v6, v6, or v4. For example:

```
SM_IP_VERSIONS=v4v6
```

4 Save and close the file.

The modified version of the runcmd_env.sh file is saved to the BASEDIR/smarts/local/conf directory.

Controlling the analysis of wide area networks

In a wide area network (WAN), where multiple logical connections emanate from a single physical interface, the IP Manager performs WAN analysis by default and diagnoses one of the following problems:

- Interface::Down, if the physical interface is operationally down, which explains why all of the subinterfaces that are layered over the physical interface have failed, along with any peer interfaces of those subinterfaces. Interface::Down alert is produced only for interfaces that do not have a discovered connection.
- Interface::LogicalConnectionDown, if two or more network connections at the subinterface level have failed and the physical interface is operationally up. “Interface” in “Interface::LogicalConnectionDown” represents the physical interface.

  Interface::LogicalConnectionDown, indicates a probable failure in the WAN network that does not cause a physical failure. The premise for the diagnosis is that a common but unobservable condition is causing all of the network connection failures.

  Whether or not the IP Manager performs WAN analysis is controlled by the SM_LOGICALCONNECTIONDOWN environment variable.

  Network connections at the subinterface level represent WAN circuits.
- NetworkConnection::Down, if just a single network connection at the interface or subinterface level has failed.

SM_LOGICALCONNECTIONDOWN environment variable

Valid values for the SM_LOGICALCONNECTIONDOWN environment variable are:

- ENABLED (default)

  The IP Manager performs WAN analysis, and generates an Interface::LogicalConnectionDown problem when two or more network connection failures occur at the subinterface level and the physical interface is operationally up.

- DISABLED

  The IP Manager does not perform WAN analysis and therefore does not generate an Interface::LogicalConnectionDown problem. Instead, when the physical interface is operationally up, the IP Manager generates a NetworkConnection::Down problem for each network connection failure that occurs at the subinterface level.

  Note Values ENABLED and DISABLED are not case-sensitive.

Setting SM_LOGICALCONNECTIONDOWN

The SM_LOGICALCONNECTIONDOWN environment variable is set to DISABLED by adding the following definition to the runcmd_env.sh file:

```
SM_LOGICALCONNECTIONDOWN=DISABLED
```

The runcmd_env.sh file in the IP Manager installation area will automatically set the SM_LOGICALCONNECTIONDOWN environment variable, along with any other environment variables in the file, for each application or utility that is started from the installation area.

Here are two reasons for setting the SM_LOGICALCONNECTIONDOWN environment variable to DISABLED:

- The evaluation of Interface::LogicalConnectionDown problems requires a considerable amount of CPU time.
- The analysis of IPSec tunnels requires the analysis of individual network connections at the subinterface level.

  To set the SM_LOGICALCONNECTIONDOWN environment variable to DISABLED:

  a Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the runcmd_env.sh file:

  ```
  sm_edit conf/runcmd_env.sh
  ```

  b Add the following definition to the file:

  ```
  SM_LOGICALCONNECTIONDOWN=DISABLED
  ```

  c Save and close the file.

  The modified version of the runcmd_env.sh file is saved to the BASEDIR/smarts/local/conf directory.

Enabling IPSec tunnel discovery and short discovery

The IP Manager discovers IPSec tunnels and Internet Key Exchange (IKE) tunnels between IPSec-enabled routers on IPv4 networks. It uses Router, Interface, and IP object types to model the tunnels.

During a full discovery, the IP Manager discovers the terminating subinterfaces (WAN endpoints) of IPSec tunnels and IKE tunnels, and layers the subinterfaces over the underlying physical interfaces. The IP Manager creates the network connections between the local and remote subinterfaces to form the IPSec and IKE tunnels.

The IP Availability Manager monitors the tunnels by monitoring the IsDownOrFlapping attribute of the WAN Interface objects that are associated with the tunnels. When the IP Availability Manager detects a change in the IsDownOrFlapping attribute of a WAN interface that is associated with a tunnel, it places the host router on the short discovery queue for a short discovery, to update the dynamic indices of the interfaces on the host router.

The parameters in Parameters for IPSec tunnel discovery and short discovery enable and control IPSec tunnel discovery and short discovery.

Table 10-2. Parameters for IPSec tunnel discovery and short discovery

Parameter name	Allowed values	Description
EnableIPSecDiscovery	TRUE,FALSE Default: FALSE Enables or disables the discovery of IPSec tunnels.	Belongs to tpmgr-param.conf file and is described in “Description of tpmgr-param.conf” on page 40.
ShortDiscoveryInstrPattern	Card_Fault_CiscoONSCPU *or* Card_Fault_CiscoEntityFRU Instrumentation class pattern to be included in short discovery.	Belongs to tpmgr-param.conf file and is described in “Other interface-limiting parameter types in the tpmgr-param.conf file” on page 126.
autoReprobe_short	TRUE,FALSE Default: FALSE Enables or disables short discovery.	Belongs to discovery.conf file and is described in “Description of discovery.conf” on page 33.
reprobePeriod_short	Integer Default: 900 seconds (15 minutes) Interval between successive short discoveries.	
numberShortProbeThreads	Integer Default: 7 threads Number of discovery threads that are used by the short discovery process.	

By default, neither IPSec tunnel discovery nor short discovery is enabled.

Also, by default, WAN analysis is enabled, as explained in *“Controlling the analysis of wide area networks” on page 34*. IPSec tunnel analysis requires that WAN analysis be disabled.

Enabling IPSec tunnel discovery and short discovery

To enable IPSec tunnel discovery and short discovery:

1 Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the tpmgr-param.conf file:

```
sm_edit conf/discovery/tpmgr-param.conf
```

2 Set the following parameter to TRUE:

```
EnableIPSecDiscovery TRUE
```

3 Save and close the file.

The modified version of the tpmgr-param.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

4 Issue the following command from *BASEDIR/smarts/bin* to make your changes to the *tpmgr-param.conf* file take effect:

```
sm_tpmgr -s <IP Manager instance name> --load-conf=tpmgr-param.conf
```

5 Type the following command from BASEDIR/smarts/bin to open the discovery.conf file:

```
sm_edit conf/discovery/discovery.conf
```

6 Set the following parameter to TRUE:

```
autoReprobe_short TRUE
```

7 Save and close the file.

The modified version of the discovery.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

8 Issue the following command from *BASEDIR/smarts/bin* to make your changes to the *discovery.conf* file take effect:

```
sm_tpmgr -s <IP Manager instance name> --load-conf=discovery.conf
```

The VMware Smart Assurance IP Manager Concepts Guide describes IPSec tunnel discovery and short discovery.

Also, if you have not already done so, consult Setting SM_LOGICALCONNECTIONDOWN and set SM_LOGICALCONNECTIONDOWN to DISABLED in the runcmd_env.sh file.

Configuration Scanner tool

The Configuration Scanner tool scans for configuration changes in your current installation. It scans for customizations with regard to the following:

- **Polling and threshold settings:** The tool presents the non-default values and settings, that is, the tool generates a list of polling settings and threshold parameters that have changed from their default values, along with details of all the groups they are associated with. The output also includes tagging settings. In case of CLI settings, the tool collects and displays the username and the associated matching criteria.
- **Configuration files:** The tool generates a list of files that have changed from the default installation. The tool scans the files in the SM_SITEMOD and base installation and does a two-way difference analysis to figure out what files have changed, and displays the list of files, flagged appropriately as modified or added. For discovery.conf, name-resolver.conf, and tpmgr-param.conf files, the tool parses through the content and presents the difference at an attribute->value pair level. The clientConnect.conf, serverConnect.conf, brokerConnect.conf, .imk.dat files, and the l10n classes and Perl directories are excluded from the scan.

Note The tool ignores service pack and hotfix files that are not modified by you.

Running the Configuration Scanner tool from the sm_getinfo utility

To run the Configuration Scanner tool:

1. Go to the <**BASEDIR**>/smarts/bin folder.

   In this document, the term **BASEDIR** represents the location where VMware Smart Assurance software is installed. For example:

   For UNIX, this location is: /opt/InCharge/<*productsuite*>.

   On UNIX operating systems, VMware Smart Assurance IP Availability Manager is, by default, installed to: /opt/InCharge/IP/smarts. This location is referred to as **BASEDIR**\smarts. Optionally, you can specify the root of **BASEDIR** to be something different, but you cannot change the <*productsuite*> location under the root directory.

2. Type the following command:

```
                sm_perl sm_getinfo <options> -k (or --smconfigscan)
sm_perl sm_getinfo --broker=localhost:5086 --server=INCHARGE-AM --smconfigscan
or
sm_perl sm_getinfo -b localhost:5086 -s pserver --smconfigscan
```

Files created by the Configuration Scanner tool

The following files are created by the tool:

- sm_configscan_report-<time_stamp>.txt: This is the report file created by the Configuration Scanner tool. This file contains:
  - List of files installed by the TTP, if any
  - List of files introduced by the user
  - Modifications made to the discovery.conf, tpmgr-param.conf, and name-resolver.conf files
  - List of file differences for each modified file
  - List of modifications found in the server
- sm_configscan-<time_stamp>.tar: This tar file contains the following:
  - A copy of the Configuration Scanner report file
  - sm_configscanner.log
  - A copy of runcmd_env.sh
  - ConfigScanAdapter.log
  - PnTallOutput.txt: lists all the settings found on the server
  - A directory containing the DIFF files for each modified files

    The output files for the configuration scanner tool is available under the

    <**BASEDIR**>/smarts/local/logs/Final_sm_getinfo<timesatamp>.tar file.

    The VMware Smart Assurance Installation Guide for SAM, IP, ESM, MPLS, and NPM Managers provides more information on the tool including sample outputs.

IP Configuration Manager

The IP-Configuration Manager is an optional component and provides a method for applying a consistent set of polling and threshold settings, IP tagging settings, and CLI device-access credentials to all IP domain managers. When the IP-Configuration Manager is not used, configure each IP domain manager with the polling and threshold settings applicable to each.

The default deployment of the IP Manager has only one IP-Configuration Manager to which any number of IP domain managers can be assigned. This works best when you deploy multiple IP domain managers with similar groups and settings.

Use the IP-Configuration Manager feature to maintain a centralized copy of the settings by attaching the Polling and Thresholds Console to the IP-Configuration Manager and pushing settings directly to those IP Managers which are “configuration-enabled”.

The “IP Configuration Manager” section in the VMware Smart Assurance IP Manager Concepts Guide provides more information on the IP Configuration Manager.

Install and start the IP-Configuration Manager

The IP-Configuration Manager server must be running before attaching the Polling and Thresholds Console. If IP-Configuration Manager was not installed as a service during installation of the IP Manager, you may install the service manually with the following command:

Unix

```
/opt/InCharge/IP/smarts/bin/sm_service install
--force
--name=ic-ip-configuration
"--name=VMware Smart Assurance IP Configuration Manager."
"--description=VMware Smart Assurance IP Configuration Manager."
--startmode=runonce
/opt/InCharge/IP/smarts/bin/sm_server.exe
"--name=INCHARGE-CM"
"--config=icf-c"
"--bootstrap=bootstrap.conf"
"--port=0"
"--subscribe=default"
"--ignore-restore-errors"
"--nodx"
"--output"
```

Detach the IP Manager from the Polling and Thresholds Console and then reattach it in order to reset the editing capability of the individual IP Manager.Apply discovery settings from the IP-Configuration Manager to IP domain managers

Enable the feature to apply discovery and auto-discovery settings from the IP-Configuration Manager to the target IP domain managers, by using this procedure:

1 Attach the Domain Manager Administration Console to the IP-Configuration Manager.

2 In the Domain Manager Administration Console, select each target IP domain manager under IC-Domain in the left panel of the console and set the attribute, **ApplyDiscoverySettings** to **TRUE**.

3 Select the IP-Configuration Manager and configure the discovery and auto-discovery settings in the **Topology** and **Dicovery Filters** tabs.

   After modification, click the Apply button in the right panel to update the settings in the IP-Configuration Manager and then click Push Settings to implement the new settings in all configuration-enabled target IP domain managers simultaneously.

Copy and customize Polling and Threshold settings in the IP-Configuration Manager

To assign one or more IP domain managers to a copied or customized Polling and Thresholds group, do the following:

1 In the Polling and Thresholds Console, right-click a Polling Group to copy (for example, Routers) and choose **Copy Configuration**. The Copy Configuration dialog box appears.

2 In the **Copy Configuration** dialog box, type the name of the new group (for example, Copy of Routers), a description, and then type the names of the IP domain managers assigned to the new group. Click **OK**.

   The new group will appear in the left panel of the console above the source group from which it was copied. This ensures that the custom group settings are applied first to the IP domain managers assigned to this group and not overwritten by the default group settings.

   Note Since the copied group has a higher priority over its source group, you will not be allowed to move the source group above the copied group.

3 Customize the settings in this new group and then push those settings to the affected IP domain managers by clicking the **Push Settings** button or by selecting it from the **Group** menu.

   Note The IP domain manager servers must be running and the attribute, *IsConfEnabled* must be *True* for each server in order for the settings to be pushed.

Display groups for a domain

The **Groups** tab for an IP domain manager attached to the IP-Configuration Manager displays all the groups that are applicable to that domain. To view the groups in the Domain Manager Administration Console, select the IP domain manager under IC-Domain and select the **Groups** tab. The groups are displayed in the format *<Parent Group Name> - <Child Group Name>*.

Through the command line, you can view the groups by typing the following dmctl command:

```
dmctl -s INCHARGE-CM invoke CM_Configuration::CM-Configuration getConfigGroups <domain_name>
```

<domain_name> is the name of the domain object in the IP-Configuration Manager topology.

Device access credentials and IP tagging settings

Device access credentials and IP tagging settings can be configured from the Polling and Thresholds Console by selecting the **Device Access** tab and **Tagging** tab respectively.

After modification, click the **Apply** button in the right panel to update the settings in the IP-Configuration Manager and then click **Push Settings** to implement the new settings in all configuration-enabled target IP domain managers simultaneously.

Configuring the Global Manager for communication with underlying domains

In addition to performing the configuration and administration tasks that are common to all Global Managers, you perform the following additional tasks to set up the Global Manager in an IP Manager deployment:

- Configure the Global Manager and any remote Global Consoles with the SM_IP_VERSIONS environment variable for name resolution and server-to-server communication.

  The VMware Smart Assurance Service Assurance Manager Configuration Guide provides information for performing this task.
- Set up a communication link between the Global Manager and the IP Manager components.
- Set up a communication link between the Global Manager and the Adapter Platform.
- Set up a communication link between the Global Manager and the satellite Domain Managers.

  The configuration guides for the individual satellite Domain Managers provide information for performing this task.

  Note Procedures for configuring all aspects of the Global Manager are provided in the VMware Smart Assurance Service Assurance Manager Configuration Guide.

Configuring Broker and server-to-server communications

Configuring Broker and server-to-server communications pertains to two areas:

- Secure connections
- IP versioning

  The VMware Smart Assurance System Administration Guide presents detailed information about VMware Smart Assurance secure communications.

Global Manager

Most of the configuration tasks for setting up the Global Manager in an IP Manager deployment are accomplished by using the Global Manager Administration Console view of the Global Console. Global Manager Administration Console interaction with the Global Manager summarizes the Global Manager Administration Console interaction with the Global Manager.

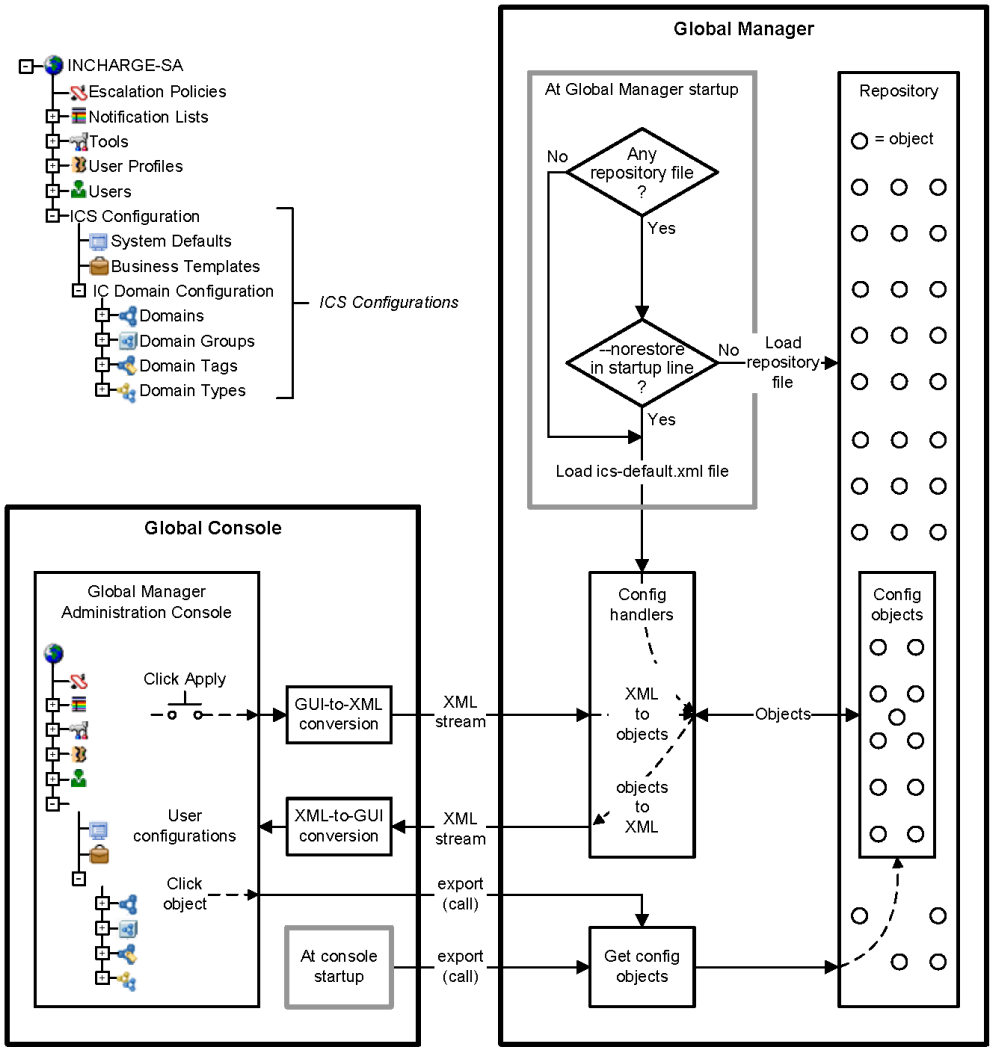


Figure 10-2. Global Manager Administration Console interaction with the Global Manager

IC Domain Configuration objects

Global Manager Administration Console presents an example display of a Global Manager Administration Console that is attached to a Global Manager named INCHARGE-SA.

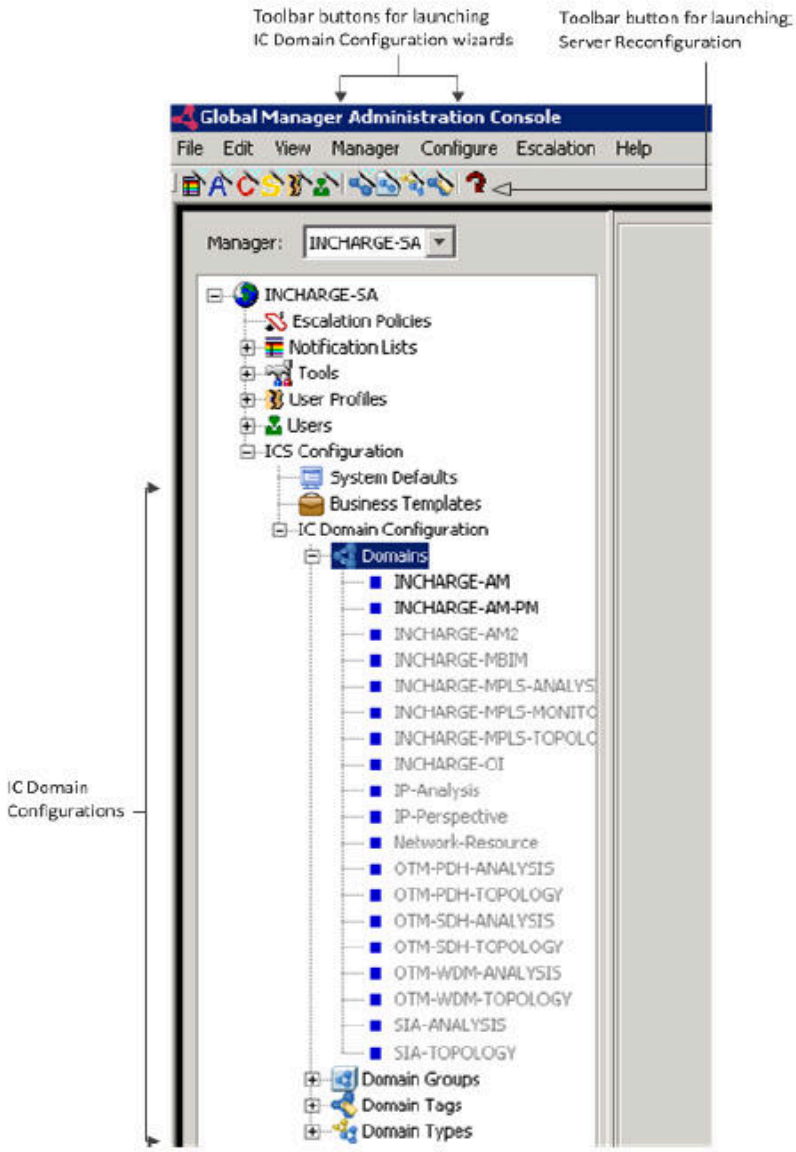


Figure 10-3. Global Manager Administration Console

Listed at the bottom of the configuration tree of the Global Manager Administration Console are the four categories of IC Domain Configuration objects: Domains, Domain Groups, Domain Tags, and Domain Types. Listed under each category are instances of these objects; for example, the default domain “INCHARGE-AM” is listed under Domains. Selecting an object in the left panel displays its attributes and their values in the right panel.

The names of enabled domains appear as black text in the Global Manager Administration Console, and the names of disabled domains appear as gray text. For example, INCHARGE-AM-PM, appears in black text because it is one of the enabled domains.

The VMware Smart Assurance Service Assurance Manager Configuration Guide provides more information on creating, modifying, and deleting existing domains.

IC Domain Configuration creation wizards

Each IC Domain Configuration category has an associated creation wizard to simplify how to specify its configuration objects. You, as an administrator, launch an IC Domain Configuration wizard in one of three ways:

- By clicking one of the four wizard toolbar buttons that are identified in IC Domain Configuration creation wizards.
- By right-clicking an IC Domain Configuration category name (for example, right-clicking Domains) and then clicking the pop-up menu
- By selecting New Domain, New Domain Group, New Domain Type, or New Domain Tag from the Edit menu

  When you create a domain, the created domain is enabled automatically.

IC Domain Configuration server reconfiguration

Clicking the Apply button, which is available to all configuration categories, writes configuration changes to the Global Manager's repository.

Clicking the Reconfigure button, which is available only to the IC Domain Configuration categories, calls a "reconfigure" method that determines which underlying domains need to be synchronized or resynchronized (that is, determines from which underlying domains topology or events need to be imported or reimported) and then initiates the needed data transfers. Calling the reconfigure method is referred to as "reconfiguring the Global Manager."

In addition to clicking the Reconfigure button, the reconfigure method can be initiated in two other ways:

- By clicking the Reconfigure toolbar button that is identified in IC Domain Configuration server reconfiguration
- By selecting Reconfigure Server from the Configure menu

Initial reconfiguration alignment

Every time that the Global Manager is restarted, it automatically initiates a server reconfiguration to align the IC domain group and domain configuration objects with the InCharge domain group and domain objects. For the initial startup, because the only enabled IC Domain Configuration objects are the INCHARGE-AM and INCHARGE-AM-PM domains, the only alignment is for these domains.

Specifying the underlying domains

You, as an administrator, use the IC Domain Configuration objects to specify the underlying IP Manager, Adapter Platform, and satellite Domain Manager components in an IP Manager deployment. Doing so provides the sources of topology and events for the Global Manager and associates each underlying component with the proper data exchange adapter (DXA) file. The Global Manager integrates and correlates the imported topology and events, and relates the data to services and customers.

To specify the underlying satellite Domain Managers, consult the appropriate Domain Manager configuration guides.

To specify the underlying components in a more complex IP Manager deployment, consult the VMware Smart Assurance Service Assurance Manager Configuration Guide.

Loading the XML configuration file for other Domain Manager s

The VMware Smart Assurance Service Assurance Manager Configuration Guide describes the procedure for using the server tools to load XML configuration files into the Global Manager.

Configuring the Adapter Platform

This section describes the various tasks associated with configuring the Adapter Platform.

The *VMware Smart Assurance Service Assurance Manager Adapter Platform User Guide* provides more information on creating, modifying, and deleting domains and domain types.

Enabling the SystemRestarted and ConfigChange events

When the IP Availability Manager detects a “system restarted” or “configuration change” condition, it creates and sends a SystemRestarted or ConfigChange trap to the Adapter Platform through the SNMP Trap Adapter that is configured as a trap receiver. The Adapter Platform, in turn, transforms the SystemRestarted or ConfigChange trap to a SystemRestarted or ConfigChange event, for import as a notification by the Global Manager.

Generation of SystemRestarted and ConfigChange events shows the events that are initiated by the IP Availability Manager for a managed network system that has been restarted or reconfigured.

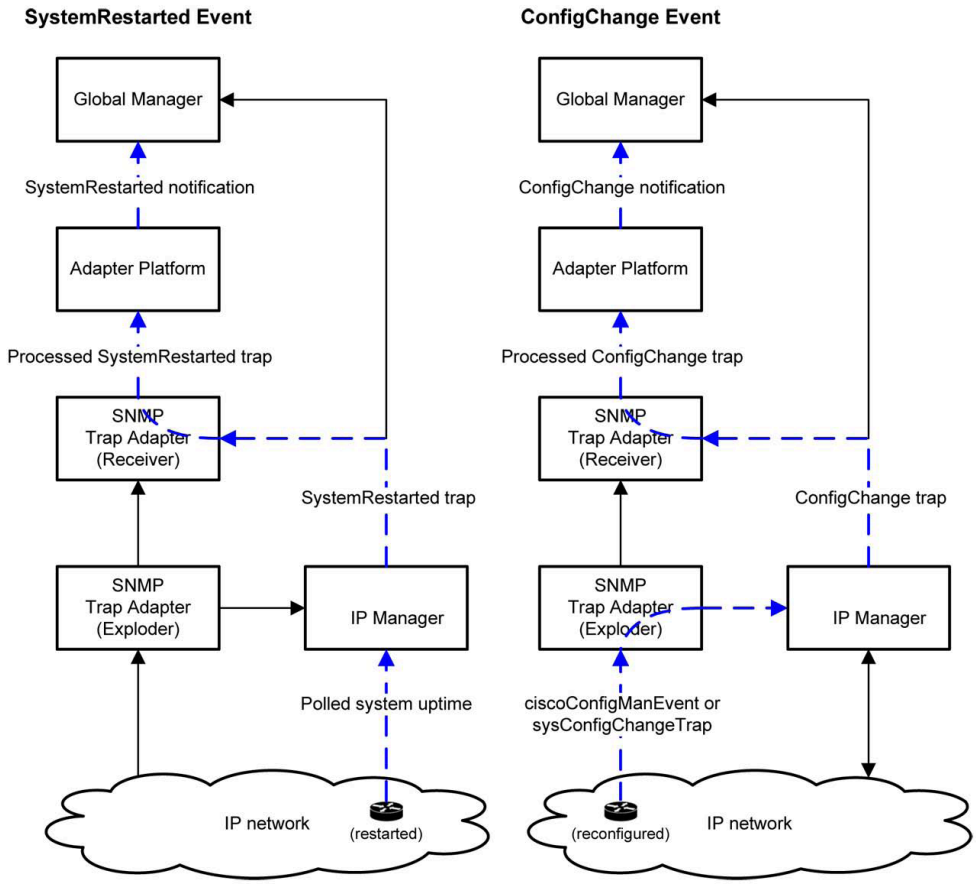


Figure 10-4. Generation of SystemRestarted and ConfigChange events

Enabling the SystemRestarted event

How the SystemRestarted event is enabled and processed shows the enabling and processing of the SystemRestarted event.

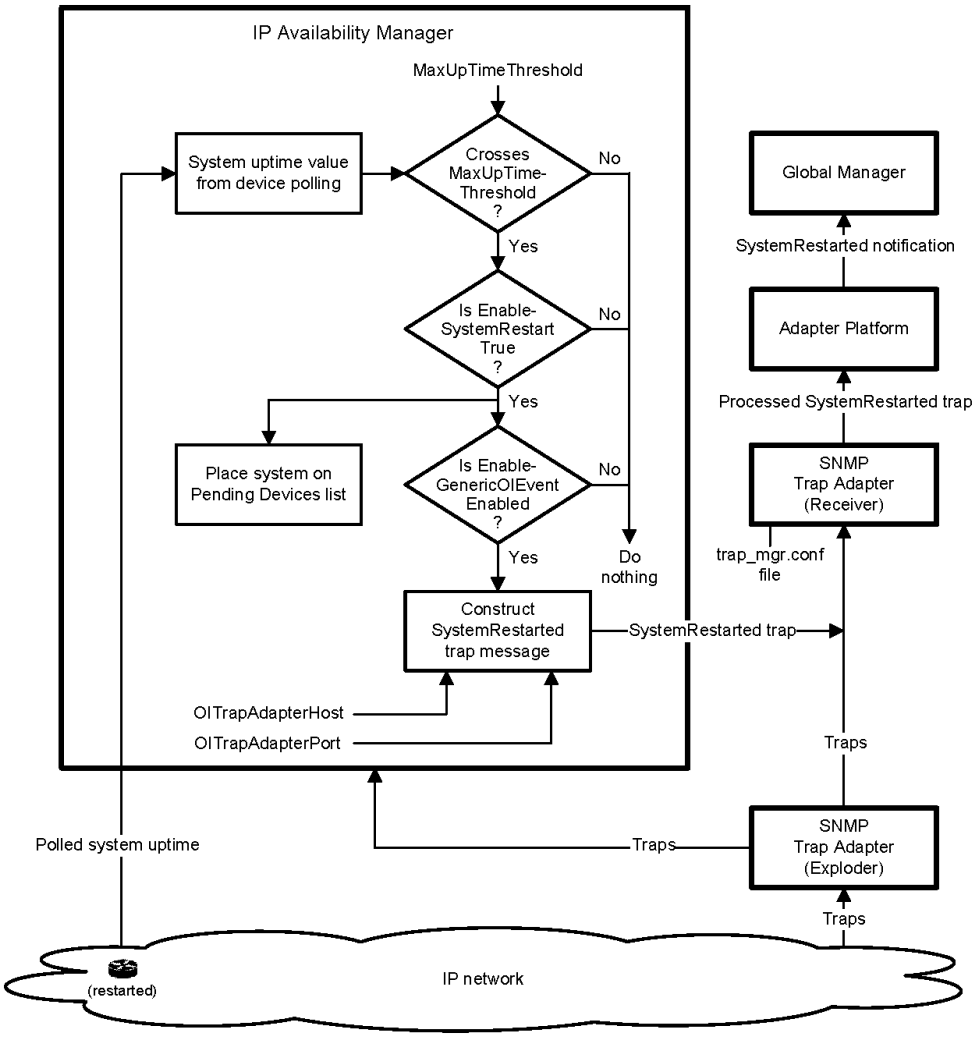


Figure 10-5. How the SystemRestarted event is enabled and processed

Assuming that the SystemRestarted trap capability is enabled, the IP Availability Manager responds as follows upon polling a system and receiving a system uptime value that crosses the MaxUpTimeThreshold value:

- Places the system on the Pending Devices list
- Generates a SystemRestarted trap for input to the Adapter Platform

  During the next pending discovery, the IP Availability Manager will send SNMP polls to the system to rediscover the system and its components.

Parameters for the SystemRestarted event

The parameters in Parameters for the SystemRestarted event enable and control the SystemRestarted event.

Table 10-3. Parameters for the SystemRestarted event

Parameter name	Allowed values	Description
EnableGenericOIEvent	ENABLED,DISABLED Default: DISABLED Controls whether a SystemRestarted trap for a system is sent to the Adapter Platform, through the SNMP Trap Adapter that is configured as a trap receiver.	Belongs to Connectivity setting and is described in “Connectivity threshold setting” on page 181.
MaxUpTimeThreshold	1 to 1436 minutes Default: 60 minutes Amount of system up time at which a system is considered to have restarted.	
OITrapAdapterHost	String Default: localhost Name or IP address of the host system where the SNMP Trap Adapter that is configured as a trap receiver is running.	Belongs to tpmgr-param.conf file and is described in “Description of tpmgr-param.conf” on page 40.
OITrapAdapterPort	Integer Default: 162 Trap listening port on the host system where the SNMP Trap Adapter that is configured as a trap receiver is running.	
EnableSystemRestart	TRUE,FALSE Default: FALSE	

As shown in Parameters for the SystemRestarted event, the polled system uptime value for a managed system is compared to the MaxUpTimeThreshold value. Whenever the system uptime value for a system equals or exceeds the MaxUpTimeThreshold value, the IP Availability Manager considers the system to have been restarted.

Trap parsing rules for the SystemRestarted event

The SNMP Trap Adapter that is configured as a trap receiver uses the rules that are specified in the trap_mgr.conf file to map a SystemRestarted trap into the data fields of a SystemRestarted notification. The Adapter Platform, in turn, creates the SystemRestarted notification (object) from the data fields and exports the notification to the Global Manager.

The parsing rules for the SystemRestarted trap are defined in the “System restart traps” section of the trap_mgr.conf file. The trap_mgr.conf file is located in the BASEDIR/smarts/conf/icoi directory of the Service Assurance Manager installation area.

Enabling the SystemRestarted event

To enable the SystemRestarted event:

1. In a Polling and Thresholds Console that is attached to the IP Availability Manager, click the **Thresholds** tab to display the thresholds group tree.

   Instructions for opening the Polling and Thresholds Console are presented in Opening the Polling and Thresholds Console.

2. In the thresholds group tree, click the plus signs (**+**) to expand the thresholds group tree to display the system resource groups: **Switches**, **Routers**, **Hubs and Bridges**, and **Other Systems**.

3. To enable the SystemRestarted event for a particular system resource group, click its **Connectivity** setting, set the **EnableGenericOIEvent** parameter to **ENABLED**, accept or modify the value of the **MaxUpTimeThreshold** parameter, and click **Apply** to save your changes.

   Procedures for modifying groups are given in Working with groups and settings.

4. Repeat step 3 for each system resource group for which you want to enable the SystemRestarted event.

5. Click **Reconfigure** to implement your changes.

6. Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the tpmgr-param.conf file:

```
sm_edit conf/discovery/tpmgr-param.conf
```

7. Set the **EnableSystemRestart** parameter to **TRUE** and accept or modify the values of the **OITrapAdapterHost** and the **OITrapAdapterPort** parameters.

8. Save and close the file.

   The modified version of the tpmgr-param.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

9. Issue the following command from BASEDIR/smarts/bin to make your changes to the *tpmgr-param.conf* file take effect:

```
sm_tpmgr -s <IP Manager instance name> --load-conf=tpmgr-param.conf
```

10 Go to the BASEDIR/smarts/bin directory in the Service Assurance Manager installation area and type the following command to open the trap_mgr.conf file:

```
sm_edit conf/icoi/trap_mgr.conf
```

11 Uncomment the following notification definition (shown uncommented) to enable the Adapter Platform to transform SystemRestarted traps into SystemRestarted notifications:

```
#System restart traps
BEGIN_TRAP .1.3.6.1.4.1.733 6 13
        ClassName:        SNMPTrap
        InstanceName:     $V1$
        ElementClassName: $V2$
        ElementName:      $V1$
        EventName:        SystemRestarted
        Severity:         3
        EventText:        System Restarted
        Expiration:       600
        State:            NOTIFY
        UnknownAgent:     CREATE
END_TRAP
BEGIN_TRAP * 0 0
        ClassName:        SNMPTrap
        InstanceName:     $SYS$
        EventName:        SystemRestarted
        Severity:         5
        Expiration:       600
        State:            NOTIFY
        EventType:        MOMENTARY
END_TRAP
BEGIN_TRAP * 1 0
        ClassName:        SNMPTrap
        InstanceName:     $SYS$
        EventName:        SystemRestarted
        Severity:         5
        Expiration:       600
        State:            NOTIFY
        EventType:        MOMENTARY
END_TRAP
```

12 Save and close the file.

The modified version of the trap_mgr.conf file is saved to the BASEDIR/smarts/local/conf/icoi directory.

13 If the Adapter Platform was running before you edited the trap_mgr.conf file, restart the Adapter Platform.

The VMware Smart Assurance Service Assurance Manager Adapter Platform User Guide provides detailed procedures for configuring the Adapter Platform and the SNMP Trap Adapter.

Enabling the ConfigChange event

How the ConfigChange event is enabled and processed shows the enabling and processing of the ConfigChange event.

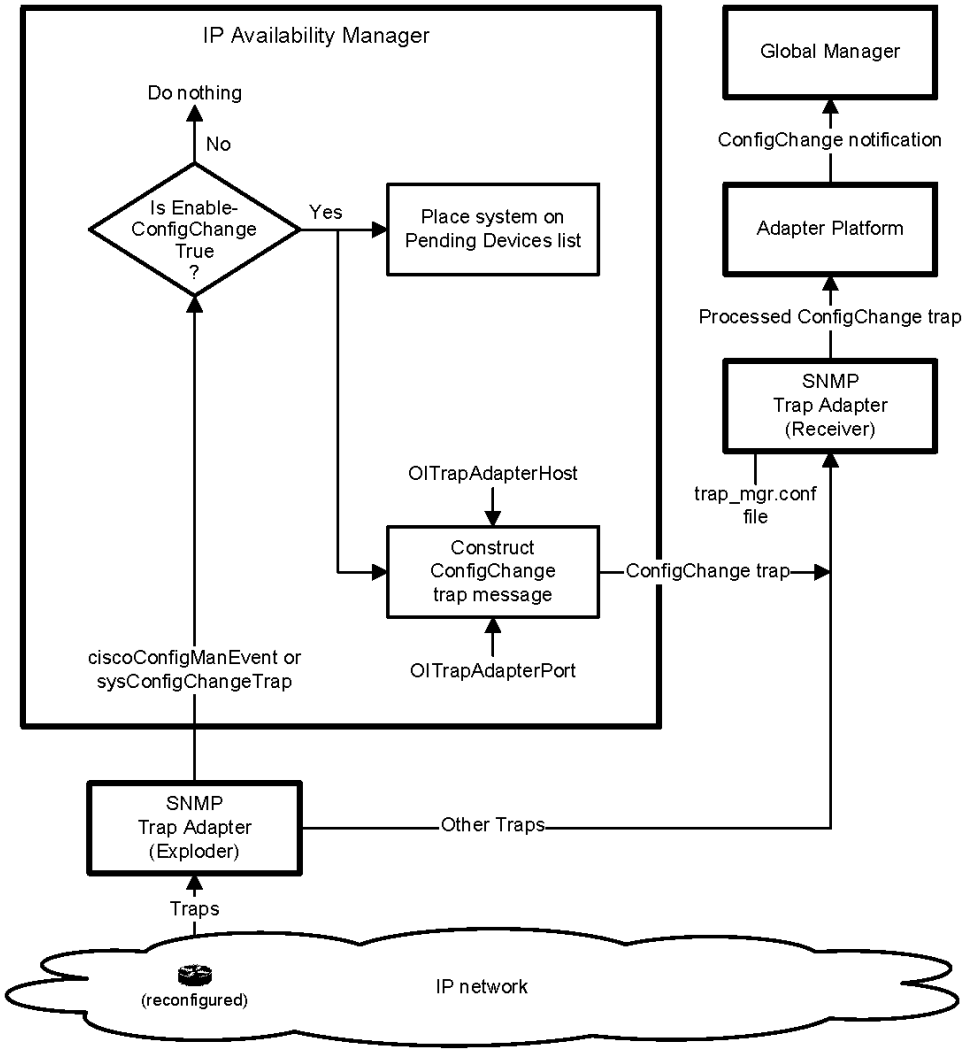


Figure 10-6. How the ConfigChange event is enabled and processed

Assuming that the ConfigChange trap capability is enabled, the IP Availability Manager responds as follows upon receiving a ciscoConfigManEvent or a sysConfigChangeTrap for a system that is already in the modeled topology:

- Places the system on the Pending Devices list
- Generates a ConfigChange trap for input to the Adapter Platform

  During the next pending discovery, the IP Availability Manager will send SNMP polls to the system to rediscover the system and its components.

Trap parsing rules for the ConfigChange event

The SNMP Trap Adapter that is configured as a trap receiver uses the rules that are specified in the trap_mgr.conf file to map a ConfigChange trap into the data fields of a ConfigChange notification. The Adapter Platform, in turn, creates the ConfigChange notification (object) from the data fields and exports the notification to the Global Manager.

The parsing rules for the ConfigChange trap are defined in the “Cisco Configuration Management Traps” and “Cisco Configuration change Traps” sections of the trap_mgr.conf file. The trap_mgr.conf file is located in the BASEDIR/smarts/conf/icoi directory of the Service Assurance Manager installation area.

Enabling the ConfigChange event

To enable the ConfigChange event:

1. Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the tpmgr-param.conf file:

   ```
   sm_edit conf/discovery/tpmgr-param.conf
   ```

2. Set the **EnableConfigChange** parameter to **TRUE** and accept or modify the values of the **OITrapAdapterHost** and the **OITrapAdapterPort** parameters.
3. Save and close the file.

   The modified version of the tpmgr-param.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.
4. Issue the following command from *BASEDIR/smarts/bin* to make your changes to the *tpmgr-param.conf* file take effect:

   ```
   sm_tpmgr -s <IP Manager instance name> --load-conf=tpmgr-param.conf
   ```

5. Go to the BASEDIR/smarts/bin directory in the Service Assurance Manager installation area and type the following command to open the trap_mgr.conf file:

   ```
   sm_edit conf/icoi/trap_mgr.conf
   ```

6 Uncomment the following notification definition (shown uncommented) to enable the Adapter Platform to transform ConfigChange traps into ConfigChange notifications:

```
# Cisco Configuration change Traps
# ------------------------------------

BEGIN_TRAP .1.3.6.1.4.1.9.5 6 9
        ClassName:        SNMPTrap
        InstanceName:     $SYS$
        EventName:        ConfigChange
        Severity:         3
        EventText:        sysConfigChangeInfo = $V2$ |
                            sysConfigChangeTime = $V1$
        Expiration:       600
        State:            NOTIFY
        EventType:        DURABLE
END_TRAP
```

7 Save and close the file.

The modified version of the trap_mgr.conf file is saved to the BASEDIR/smarts/local/conf/icoi directory.

8 If the Adapter Platform was running before you edited the trap_mgr.conf file, restart the Adapter Platform.

The VMware Smart Assurance Service Assurance Manager Adapter Platform User Guide provides detailed procedures for configuring the Adapter Platform and the SNMP Trap Adapter.

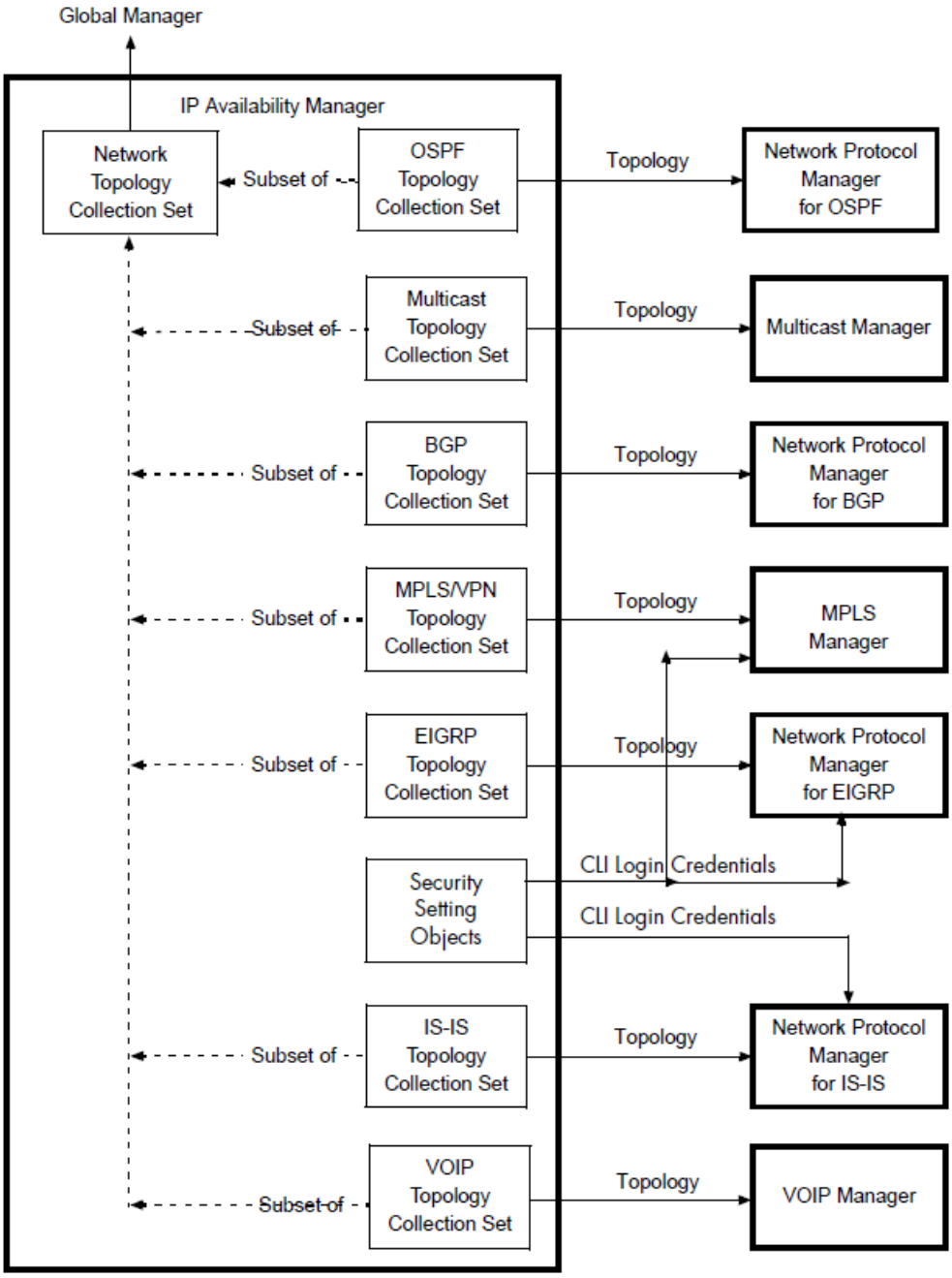
Configuring IP to enable light discovery for Satellite Domain Manager

This section describes the various tasks associated with configuring IP to enable light discovery for satellite Domain Manager.

Enabling light discovery for satellite Domain Manager

By default, the IP Availability Manager is not configured to discover the topology collection sets for the satellite Domain Managers that are shown in Exporting topology collection sets to satellite Domain Managers. The creation of topology collection sets is known as “IP Manager light discovery” or simply “light discovery.”

Figure 10-7. Exporting topology collection sets to satellite Domain Manager



The term “satellite,” as used here, refers to any of the VMware Smart Assurance Domain Managers that import topology and topology updates from the IP Manager.

To configure IP Availability Manager to discover the topology collection set for MPLS Manager, the following parameter must be set to FALSE:

```
DisableMPLSDiscovery FALSE
```

To configure IP Availability Manager to discover the topology collection set for Multicast Manager, the following parameter must be set to FALSE:

```
DisableMulticastDiscovery FALSE
```

To configure IP Availability Manager to discover the topology collection set for Network Protocol Manager, the following parameter must be set to FALSE:

```
DisableProtocolDiscovery FALSE
```

To configure IP Availability Manager to discover the topology collection set for VoIP Availability Manager, the following parameter must be set to FALSE:

```
DisableVOIPDiscovery FALSE
```

All four parameters are in the tpmgr-param.conf file, which is described in “Description of tpmgr-param.conf” on page 40.

Enabling light discovery for satellite Domain Manager

To configure IP Availability Manager to discover the topology collection sets for MPLS Manager, Multicast Manager, Network Protocol Manager, and VoIP Availability Manager:

1 Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and issue the following command to open the tpmgr-param.conf file:

```
sm_edit conf/discovery/tpmgr-param.conf
```

2 Set the following four parameters to FALSE:

```
DisableMPLSDiscovery FALSE
DisableMulticastDiscovery FALSE
DisableProtocolDiscovery FALSE
DisableVOIPDiscovery FALSE
```

3 Save and close the file.

The modified version of the tpmgr-param.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

4 Issue the following command from *BASEDIR/smarts/bin* to make your changes take effect:

```
sm_tpmgr -s <IP Manager instance name> --load-conf=tpmgr-param.conf
```

Or, as an alternative to the preceding procedure, use the **dmctl** utility to temporarily set these parameters to FALSE:

Go to the *BASEDIR/smarts/bin* directory in the IP Manager installation area and issue the following commands:

```
dmctl -s <IP Manager instance name> invoke ICF_TopologyManager::ICF-TopologyManager
insertParameter DisableMPLSDiscovery FALSE
dmctl -s <IP Manager instance name> invoke ICF_TopologyManager::ICF-TopologyManager
insertParameter DisableMulticastDiscovery FALSE
dmctl -s <IP Manager instance name> invoke ICF_TopologyManager::ICF-TopologyManager
insertParameter DisableProtocolDiscovery FALSE
```

Example of configuring interface-matching filters

This section presents an example of configuring interface-matching filters that will exclude any interface that has “Ethernet” in its description during the discovery of an Adtran Atlas 830 Router device, which has a sysObjectID of .1.3.6.1.4.1.664.1.430.

Be aware that your matching patterns and the retrieved interface descriptions are not case-sensitive because the IP Manager transforms both of them to uppercase letters before comparing them with one another.

To configure the interface-matching filters for the Adtran Atlas 830 Router device:

1 Perform a MIB walk of the device to obtain a list of interfaces for the device.

2 Inspect the interfaces and attempt to form a matching pattern for the interfaces that you want to exclude from the discovered topology.

   For example, look at their IfType (.1.3.6.1.2.1.2.2.1.3) values and ifDescr (.1.3.6.1.2.1.2.2.1.2) values and determine if the interfaces that you want to exclude all have the same type or all have the same string in their descriptions.

   In this procedure, the matching pattern “~*Ethernet*” will be used, a pattern that appears in interface ifDescr descriptions such as “Ethernet0” and “FastEthernet1/0.”

3 To obtain the values (interface-matching patterns) for the device, go to the BASEDIR/smarts/bin directory of the IP Manager installation area and type the following commands:

```
dmctl “s <

   IP Manager

 instance name
> invoke

   ICF_TopologyManager::ICF-TopologyManager findParameter

      IFIndexPattern-.1.3.6.1.4.1.664.1.430
dmctl “s <

   IP Manager
```

```
 instance name>
 invoke

   ICF_TopologyManager::ICF-TopologyManager findParameter

      IFDescrPattern-.1.3.6.1.4.1.664.1.430
dmctl "s <

   IP Manager

 instance name
> invoke

   ICF_TopologyManager::ICF-TopologyManager findParameter

      IFTypePattern-.1.3.6.1.4.1.664.1.430
```

4 Whatever values you start with (the default values or the values obtained from the dmctl commands), change those values to exclude the interfaces that have "Ethernet" in their descriptions.

The following values will be used in this example:

- IFIndexPattern-.1.3.6.1.4.1.664.1.430 ~*
  - IFDescrPattern-.1.3.6.1.4.1.664.1.430 ~*Ethernet*
  - IFTypePattern-.1.3.6.1.4.1.664.1.430 ~*

    No interface descriptions can match the "~*" value, and only interface descriptions that do *not* contain the string "Ethernet" can match the "~*Ethernet*" value.

5 Type the following command to open the tpmgr-param.conf file:

```
sm_edit conf/discovery/tpmgr-param.conf
```

6 Add the following two lines to the file:

```
IFDescrPattern-.1.3.6.1.4.1.664.1.430 ~*Ethernet*
IFTypePattern-.1.3.6.1.4.1.664.1.430 ~*
```

7 Save and close the file.

The modified version of the tpmgr-param.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

8 To force the ICF-TopologyManager to read the tpmgr-param.conf file, type the following command:

```
sm_tpmgr -s <

   IP Manager

 instance name
> --load-conf=tpmgr-param.conf
```

9 To confirm that your configuration has been read by the ICF-TopologyManager, type the following commands:

```
dmctl -s <

   IP Manager

 instance name
> invoke

   ICF_TopologyManager::ICF-TopologyManager findParameter

     IFDescrPattern-.1.3.6.1.4.1.664.1.430
dmctl -s <

   IP Manager

 instance name>
 invoke

   ICF_TopologyManager::ICF-TopologyManager findParameter

      IFTypePattern-.1.3.6.1.4.1.664.1.430
```

10 Examine the returned values and confirm that they are the same ones added to the tpmgr-param.conf file in step 6.

Note The following steps verify that the interface-matching filters are configured properly.

11 Attach the Domain Manager Administration Console to the IP Manager.

Instructions for opening the Domain Manager Administration Console are presented in Chapter 3, “Opening the Global Console in the section Opening the Global Console.

12 Delete every instance of the Adtran Atlas 830 Router device in the topology tree. To delete a device, right-click its name and select **Delete**.

When you delete a device, all of its components (including interfaces) and all of its connections are also deleted.

13 Rediscover the device by selecting **Topology** > **Add Agent and specifying the hostname or IP address of the device.** Or create a seed file and import it by using **Topology** > **Import from seed file**.

14 Expand the **Router** class in the topology tree to see the Adtran Atlas 830 Router instances that the IP Manager just rediscovered.

15 Verify in the **Attributes** tab that the Description attribute does not contain the “Ethernet” string.

Optimizing the discovery of ERX devices

The discovery of Juniper ERX devices that are running JUNOS 6.x and higher might take longer than what is acceptable because the IP Manager, by default, discovers all interfaces on an ERX device. Depending on the deployment, discovering all interfaces on an ERX device is not always desirable, especially when speed of discovery is an issue.

Note The term “Juniper ERX device” or “ERX device,” as used here, represents an instance of any discovered Juniper ERX main or virtual router.

The solution to this problem is to use the parameters to configure the IP Manager to discover fewer interfaces on the discovered ERX devices. “Optimizing the discovery of ERX devices” table in Chapter 6, Configuration Settings, in the VMware Smart Assurance IP Manager Reference Guide lists the parameters for optimizing the discovery of ERX devices. The solution is based on the filtering of ERX device interfaces by type and description to prevent (exclude) undesirable interfaces from being discovered.

The parameters are described in the *Other interface-limiting parameter types* table in the VMware Smart Assurance IP Manager Reference Guide.

Optimizing the discovery of ERX devices

To optimize the discovery of Juniper ERX devices:

1 Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the tpmgr-param.conf file:

```
sm_edit conf/discovery/tpmgr-param.conf
```

2 Set all or any of the following parameters to optimize the discovery of Juniper ERX devices in your environment:

```
ERXIfExcludeSysPattern
ERXIfExcludeTypePattern
ERXIfExcludeDescrPattern
IFTypePattern
IFDescrPattern
```

3 Save and close the file.

   The modified version of the tpmgr-param.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

4 Type the following command from BASEDIR/smarts/bin to implement your changes:

```
sm_tpmgr -s <
   IP Manager
 instance name
>
   --load-conf=tpmgr-param.conf
```

Enabling the creation of a physical connection for a shared MAC address

The IP Manager perceives each interface found in the ifTable of a discovered device as being a physical interface, meaning that the IP Manager does not distinguish between physical and virtual interfaces. Thus, when the IP Manager detects multiple interfaces (one physical and the others virtual) that share the same MAC address, it perceives the condition as an error and subsequently removes the “LayeredOver/Underlying” relationships between the interfaces and the MAC address—relationships that the IP Manager created initially for each interface that it found in the ifTable. Without a LayeredOver/Underlying relationship between the physical interface and the MAC address, the IP Manager will not create the network connection for the physical interface during the postprocessing phase of the discovery process.

By identifying the virtual interfaces for a device in the IfVirtualTypePattern parameter, a user changes the behavior of the IP Manager in the following way: The IP Manager will not create LayeredOver/Underlying relationships between these interfaces and the underlying MAC address. Accordingly, the LayeredOver/Underlying relationship between the physical interface and the underlying MAC address will not be removed, and the network connection for the physical interface will be created during the postprocessing phase of the discovery process.

The IfVirtualTypePattern parameter is described in the *Other interface-limiting parameter types* table in the VMware Smart Assurance IP Manager Reference Guide.

Enabling the creation of a physical connection for a shared MAC address

To enable the IP Manager to create a physical connection for a shared MAC address:

1 Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the tpmgr-param.conf file:

```
sm_edit conf/discovery/tpmgr-param.conf
```

2 Set the following parameter to the sysObjectID of the target device and to the ifType values of the target virtual interfaces:

```
IfVirtualTypePattern-<sysObjectID> <interface type pattern>
ifVirtualTypePattern-.1.3.6.1.4.1.9.1.452 135|136
```

3 Save and close the file.

The modified version of the tpmgr-param.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

4 Type the following command from BASEDIR/smarts/bin to implement your changes:

```
sm_tpmgr -s <

   IP Manager

 instance name
>

   --load-conf=tpmgr-param.conf
```

5 Press **Enter**.

Running IP server as non root on UNIX and Linux systems

Options for running as non root on UNIX and Linux systems describes two options that are used in conjunction with each other to run IP server as a server or as a service, as non root. These two options are only available on UNIX and Linux systems. These options are available for sm_server.

Table 10-4. Options for running as non root on UNIX and Linux systems

Options	Descripiton
--privopen=<arg>[,<*arg*>]	**Note** This option is only used in conjuction with the --run-as-user option to run as non root. Opens privileged sockets. The <*arg*> parameter has the following syntax: <type>[:<*family*>]:<*port*>[,<*count*>] where: ■ <*type*> is one of the following: ■ TCP (for a TCP connection) ■ UDP (for a datagram) ■ IP (for a raw socket) ■ <*port*> is one of the following: ■ The required privileged port for a TCP socket ■ The required privleged port for a UDP socket ■ The protocol for IP ■ <*family*> is one of the following: ■ :v4 (IPv4 address family) ■ :v6 (IPv6 address family) ■ If <*family*> is not specified, the address family defaults to IPv4. ■ <*count*> is the number of sockets of the type, family and port. The default count is 1.
--run-as-user=<*username*>	**Note** This option is only used in conjuction with the --privopen option to run as non root. Specifies a valid user name.

The --privopen and --run-as-user options are used in conjunction to run a Domain Manager with non root privileges. The Domain Manager can startup as root, open privileged ports, and change to a non root user. You can open multiple ports by repeating the <arg> parameter. Each instance is separated by a comma (,). For example:

```
--privopen=UDP:v4:161,#2
--privopen=IP:1,#4
--privopen=IP:v6:58
--run-as-user=testuser1
```

In this example:

- The first privopen line opens two UDP IPv4 sockets on port 161.
- The second privopen line opens four raw IPv4 sockets for ICMP.
- The third privopen line opens one raw IPv6: socket for ICMP.

  After the sockets are opened, the process will change to run as user, "testuser1".

  Note The following privileged port is required by the sm_server:

  port 162 (to listen to incoming traps).

Using --privopen and --run-as-user when starting sm_server as a service

```
/opt/InCharge/IP/smarts/bin/sm_service install
--force
--name=ic-am-pm-server
"--description= VMware Ionix IP Availability Manager and
Performance Manager Server"
--startmode=runonce
/opt/InCharge/IP/smarts/bin/sm_server
--name=INCHARGE-AM-PM
--config=icf
--bootstrap=bootstrap-am-pm.conf
--port=0
--subscribe=default
--ignore-restore-errors
--output
--privopen=IP:v4:1,#6
--privopen=IP:v6:58,#6
--run-as-user=testuser1
```

where, *--privopen=IP:v4:1,#6* and *--privopen=IP:v6:58,#6* opens 6 RAW ICMP sockets and optionally 6 RAW ICMPv6 sockets for ICMP accessor.

Using --privopen and --run-as-user when starting sm_server as a server

```
./sm_server
--name=<server_name>
--config=icf
--bootstrap=bootstrap-am-pm.conf
--port=0
--ignore-restore-errors
--output
--privopen=IP:v4:1,#6
--privopen=IP:v6:58,#6
--run-as-user=testuser1
```

where, *--privopen=IP:v4:1,#6* and *--privopen=IP:v6:58,#6* opens 6 RAW ICMP sockets and optionally 6 RAW ICMPv6 sockets for ICMP accessor.

Upgrade and migration impacts

The “Performing Migration” chapter in the
VMware Smart Assurance Installation Guide for SAM, IP, ESM, MPLS, and NPM Managers provides information on migration procedures.

Deployment Utility

The Deployment Utility allows you to deploy customizations and configuration changes from an existing installation to another installation on the same version of a product. The deployment can be carried out on the same host or between two different hosts running the same operating system. The utility is useful if you have to apply the same configurations on mutiple installations of a product.

Use the utility if you are:

- Running multiple new installations on the same or multiple hosts that will need to share the same configuration and customizations or
- Upgrading multiple installations on the same or multiple hosts that will share the same configuration and customizations

  In both cases, start with one installation where you make all your modifications to the files and then run the sm_deploy utility to create a package that contains your modifications. Then use the sm_deploy utility to apply the files collected from the first installation to the rest of your installations.

  The Deployment Utility performs the following three functions:

- Create a package
- Deploy the package
- Rollback

  The Deployment Utility can be run using the following command:

  ```
  sm_perl sm_deploy.pl <options>
  ```

  To run the utility from an older version of a Smarts product that does not have the migration utility, copy the sm_deploy.pl script and certain Perl files (packaged in migratePerlPkg.zip file) and place them in the appropriate locations. The sm_deploy.pl must be placed in the bin directory and the Perl package must be placed under the BASEDIR/smarts/local folder and extracted there.

  The VMware Smart Assurance Installation Guide for SAM, IP, ESM, MPLS, and NPM Managers provides more information on the functions, command-line options, and how to create and deploy a deployment package.

Discovery

11

This chapter includes the following topics:

- Discovery overview
- Preparing for discovery
- Discovery options
- Initiating discovery
- Preparing for manual discovery
- Preparing for autodiscovery
- Scheduling automatic discovery
- Starting discovery
- Creating discovery filters
- OID Info Tool
- Seed file related tasks
- Discovery and short discovery of IPSec tunnels
- Customizing discovery

Discovery overview

The discovery process is anchored by Topology Manager, a singleton class of type ICF_TopologyManager which provides the interface for discovering topology elements. TopologyManager has a pending list that holds items that should be discovered in the next iteration of discovery.

This section provides the steps of discovery.

1. Attach the Global Console to the Global Manager and configure the Global Manager to import topology and events from the underlying IP Manager, satellite Domain Manager and Adapter Platform as described in Configure the Global Manager for discovery.
2. Configure the IP manager for discovery, as described in Configure the IP Manager for discovery.
3. Enable discovery features, which is discussed in Discovery options.

4 Use the Name resolution service to translate hostnames to IP addresses, and to translate IP addresses to hostnames, described in Chapter 14 Name Resolution.

5 Choose discovery address preference in the discovery.conf file:

- IPv6FIRST_IPv4NEXT
  - IPv4FIRST_IPv6NEXT
  - IPv6ONLY
  - IPv4ONLY

6 Configure full discovery interval or pending discovery interval. Add autodiscovery filters, if applicable. Scheduling automatic discovery describes the procedure.

7 Choose the discovery mode:

- Autodiscovery
  - Manual discovery
  - Light discovery

Note Besides performing a traditional discovery of network objects, you can configure the IP Manager to create topology collection sets from the discovered network objects. The creation of topology collection sets is known as “IP Manager light discovery” or simply “light discovery.” By default, light discovery is disabled. The Chapter 10 Configuration provides instructions on configuring the IP Manager to create topology collection sets.

8 Initiate the discovery process, which could be either manual discovery or autodiscovery. It involves importing a seed file or adding seed systems to the topology. This is discussed in *“Initiating discovery” on page 77*.

9 Start discovery in any one of the following ways:

- *“Starting discovery from the console” on page 87*
  - *“Starting discovery from the command line” on page 88*

    You may choose to start full discovery or start pending discovery, depending on your requirement.

Preparing for discovery

Preparing an IP Manager deployment for discovery requires configuring all component applications in the deployment so that they can form the proper physical and logical connections to one another. Some of these tasks are performed before the components are started, such as the editing of configuration files, while others are performed through the Global Console when the components are up and running.

- Configure the Global Manager for discovery
- Configure the IP Manager for discovery
  - How to create a seed file

- *"Preparing for autodiscovery" on page 81*

Configure the Global Manager for discovery

Preparing the Global Manager for discovery involves:

- *"Opening the Domain Manager Administration Console" on page 18*
- Configuring the Global Manager to import topology and events from the underlying IP Manager, Adapter Platform, and satellite Domain Manager components in an IP Manager deployment.

  An administrator configures the Global Manager to import topology and events from the IP Manager. The *VMware Smart Assurance Service Assurance Manager Operator Guide* provides instructions for preparing the Global Manager for discovery in an IP Manager deployment.

Configure the IP Manager for discovery

Preparing the IP Manager for discovery starts with seed systems, which serve as the starting point for both manual discovery and autodiscovery.

In addition, to support satellite Domain Managers, an administrator performs the following configuration tasks to prepare an IP Manager for discovery as listed in Configuration tasks to prepare an IP Manager for discovery .

Configuration tasks to prepare an IP Manager for discovery lists the configuration tasks to prepare an IP Manager for discovery.

Table 11-1. Configuration tasks to prepare an IP Manager for discovery

Configuration tasks	Applicability	Note
Enable light discovery Enabling light discovery for satellite Domain Manager describes the procedure.	Satellite Domain Manager	By default, light discovery is disabled.
Set the *DiscoveryAddrPref* to IPV4_ADDRONLY in discovery.conf file	All satellite Domain Managers	Ensures that the SNMPAgent objects that are imported by a satellite Domain Manager from the IP Manager have only IPv4 addresses.
Specify one or more CLI device-access objects	If a satellite Domain Manager is to import CLI device-access objects from the IP Manager.	An administrator can attach the Polling and Thresholds Console directly to an IP Manager to configure CLI device-access objects.
Enable overlapping IP address discovery	If a satellite Domain Manager is to import tagged IP and IPNetwork objects from the IP Manager.	NA
Enable interoperability with the VPN-tagging server	Only if the IP Manager is enabled to perform overlapping IP address discovery.	NA

The tpmgr-param.conf file contains configuration parameters that you edit to customize IP Manager discovery. The parameters are a subset of table entries that are defined for the ICF-TopologyManager object. Use the parameters in this file to accomplish such tasks as disabling the discovery of certain interfaces or ports, enabling the discovery of topology collection sets for satellite Domain Managers, and changing the operation of the IP Manager to be backward compatible with earlier versions of satellite Domain Managers. To open and edit the tpmgr-param.conf file:

1 Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the tpmgr-param.conf file:

```
sm_edit conf/discovery/tpmgr-param.conf
```

2 Press **Enter**. This opens the tpmgr-param.conf file. Make the necessary edits.

Discovery methods and the associated tasks lists the discovery methods, the scenarios when you would use it and the various tasks related to it.

Table 11-2. Discovery methods and the associated tasks

Discovery method	What it does...	Use it when...	Requirements	Tasks
Manual discovery	The IP Manager discovers a set of systems that are specified in a seed file, or discovers an individual system that is specified in an **Add Agent** command.	The topology information is available from a well-maintained database.	Seed files	■ How to create a seed file Also, see ■ *"Seed file related tasks" on page 95*
Autodiscovery	The IP Manager takes IPv4 addresses that are obtained from discovered systems to find additional discovery candidate systems.	The topology information is incomplete or unavailable, or when manual updates cannot be maintained.	Discovery filters	■ How to create a discovery filter ■ How to delete a discovery filter

Discovery options

Discovery options lists the various features that you may want to enable for discovery.

Table 11-3. Discovery options

Features	Purpose	
Incremental reconfigure	Opt for incremental reconfigure only for newly added devices, which is faster.	Enable Incremental reconfigure
Default IP route	To allows routers which are not able to maintain a huge list of routing entries in their routing table to use the default traffic to the advertised border router. You do not have to configure the static routes on the routers which are not able to maintain the routing table entries.	Enable Default IP route discovery and monitoring

Table 11-3. Discovery options (Continued)

Features	Purpose	
LLDP (Link Layer Discovery Protocol)	To allow the discovery of LLDP enabled devices by IP Availability Manager. NetworkConnections, TrunkCable, and Cable connections are created in the IP Availability Manager which processes the global data structure based on data gathered during discovery.	Enable LLDP (Link Layer Discovery Protocol)
Wireless LAN discovery	To set the flags for wireless LAN discovery	Table 7, “Wireless LAN settings,”
Virtual Port Channel	To enable/disable vPC discovery	Enabling vPC discovery
	To gather vPC data from Nexus switches.	Viewing vPC information
Virtual Data Context	To enable virtual data context discovery.	Enabling VDC discovery
	To view information about other VDCs created by the default VDC.	Viewing VDC information
Link Aggregation and Etherchannel discovery	To identify AggregatePorts and figure out ports that are part of an aggregation	Identifying AggregationPorts through CLI

Enable Incremental reconfigure

To enable incremental reconfigure, set the *ThresholdOfChangedUCSForIncReconfig* parameter value in the tpmgr-param.conf file. The value of this flag controls the maximum number of devices for which incremental reconfigure will run. For example, if the value is 10, incremental reconfigure will run if the number of the last discovered device is less then 10, otherwise a bulk reconfigure will run. This parameter is not used if reconfigure is invoked from the console. An incremental reconfigure happens only for the devices added, and is faster.

Note By default, the value of the parameter *ThresholdOfChangedUCSForIncReconfig* is set to 10 in tmpgr-param.conf. This means that for any discovery containing less than 10 systems, *Reconfigure started* and *Reconfigure finished* message will not be logged in the domain log, even though the reconfigure occurs.

Viewing Incremental reconfigure messages

To view incremental reconfigure messages, set *DebugReconfigure* to TRUE in tmpgr-param.conf.

Editing tmpgr-param.conf

1 To make the change in tmpgr-param.conf, execute the following command:

```
<BASEDIR>/IP/smarts/bin/sm_edit conf/discovery/tpmgr-param.conf
```

2 Restart the IP domain or reload tmpgr-param.conf for the change to take effect:

```
<BASEDIR>/IP/smarts/bin/sm_tpmgr -s <IP 8.1.X domain> --load-conf=tpmgr-param.conf
```

Verifying the ThresholdOfChangedUCSForIncReconfig setting

To verify the ThresholdOfChangedUCSForIncReconfig setting, execute the following command:

```
<BASEDIR>/IP/smarts/bin/dmctl -s <IP 8.1.X domain> invoke ICF_TopologyManager::ICF-TopologyManager
findParameter ThresholdOfChangedUCSForIncReconfig
```

To make change without restarting the IP domain, execute the following command:

```
<BASEDIR>/IP/smarts/bin/dmctl -s <IP 8.1.X domain> invoke ICF_TopologyManager::ICF-TopologyManager
insertParameter ThresholdOfChangedUCSForIncReconfig <value>
```

Verifying the debug settings

To verify debug setting, execute the following command:

```
<BASEDIR>/IP/smarts/bin/dmctl -s <IP 8.1.X domain> invoke ICF_TopologyManager::ICF-TopologyManager
findParameter DebugReconfigure
```

To make change without restarting the IP domain, execute the following command:

```
<BASEDIR>/IP/smarts/bin/dmctl -s <IP 8.1.X domain> invoke ICF_TopologyManager::ICF-TopologyManager
insertParameter DebugReconfigure TRUE
```

Enable Default IP route discovery and monitoring

A new setting *Default IP_Route Connectivity Polling* is added under the Routers category in the Polling tab of the Polling and Thresholds Console. By default, this setting is available for all routers and when the AnalysisMode for this setting is enabled, a new class *IP_Route* gets created.

The new class has a HostedBy relationship with the router and the new class monitors the IP route default entry. Whenever the entry is not present, that is, the 0.0.0.0 value is not there in the MIB entry, the *MissingDefaultRoute* event is generated.

Enable LLDP (Link Layer Discovery Protocol)

To do a field certification and validate the LLDP connections, invoke the LLDP framework by specifying the NEIGHBOR probe as LLDP in the configuration file *oid2type_<vendor>.conf* file.

Note “Link Layer Discovery Protocol” section in Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide, provides more information about LLDP.

Configure wireless LAN settings

Short discovery is supported for the discovery and monitoring of a Wireless LAN. Short discovery handles wireless topology changes, which typically occur during WirelessClient movement. In addition, it dynamically detects Rogue Wireless APs. Wireless LAN settings lists the flags that determine the settings for Wireless LAN discovery.

Table 11-4. Wireless LAN settings

Flag	Allowed value	Description	File
WIFI_autoReprobe_Short **Note** Server restart is required.	Default value: FALSE ■ TRUE — Enables WIFI related short discovery. ■ FALSE— Disabled WIFI related short discovery.	Enables or disables WIFI related short discovery.	discovery.conf
WIFI_reprobePeriod_short	Default value:900 seconds **Note** Value should be positive and greater than 300.	Sets the interval between successive WIFI short discoveries. **Note** Decreasing the discovery interval will increase the CPU usage. If you decrease the short discovery interval, use small decrements and monitor your current CPU usage.	tpmgr-param.conf
EnableRogueAPShortDiscovery	Default value: TRUE	Determines the discovery of Rogue AP. Disabling this flag will stop the short discovery of Rogue AP.	tpmgr-param.conf
EnableWirelessClientDiscovery	Default value: TRUE	Determines the discovery of the WirelessClient.	tpmgr-param.conf
EnableWirelessClientShortDiscovery	Default value: TRUE	Determines the short discovery of the WirelessClient	tpmgr-param.conf

Enable Virtual Port Channel settings

Enabling vPC discovery

The flag *EnableVPCDiscovery* controls vPC discovery. By default, vPC discovery is enabled in IP 9.1.

To disable vPC discovery, set the value of the flag *EnableVPCDiscovery* in <*BASEDIR*>/conf/tpmgr-param.conf file.

```
EnableVPCDiscovery= FALSE
```

For information on the events and root-cause analysis of vAggregatePort, refer to VMware Smart Assurance IP Manager Reference Guide.

Viewing vPC information

To view the information about the virtual port channels (vPCs), run the command on the Nexus switch on which vPC is configured:

```
show vpc brief
```

The command displays the following information:

- vPC domain ID
- Peer-link status
- Keepalive message status
- Configuration consistency status
- Number of vPC confgured

Note This command is available only if the vPC feature is enabled.

Sample output of the command

```
switch(config)# show vpc brief
Legend:
                (*) - local vpc is down, forwarding via vPC peer-link
vPC domain id                   : 10
Peer status                     : peer adjacency formed ok
vPC keep-alive status           : peer is alive
Configuration consistency status: success
vPC role                        : primary
Number of vPC configured        : 1
vPC Peer-link status
---------------------------------------------------------------------
id   Port   Status Active vlans
--   ----   ------ -------------------------------------------------
1    Po10   up     1-100
vPC status
----------------------------------------------------------------------------
id   Port   Status Consistency Reason                     Active vlans
--   ----   ------ ----------- -------------------------- ------------
20   Po20   up     success     success                    1-100
```

Enabling VDC discovery

The flag EnableVirtualDataContextDiscovery in tpmgr-param.conf file controls VDC discovery.

Default Value=TRUE

Disabling this flag will stop the Virtual Data Context (VDC) discovery in Nexus 7000 switches.

Note Provide the read-only or Administrator VDC credentials under the CLI Access Group in Polling and Threshold > Device Access.

Viewing VDC information

While discovering the default VDC, information about other VDCs created by the default VDC can be viewed by running the CLI command:

```
show vdc
```

A sample output of the command:

```
Router# show vdc
vdc_id  vdc_name                          state               mac
------  --------                          -----               ----------
1       switch                            active              00:18:ba:d8:3f:fd
2       Payroll                           active              00:18:ba:d8:3f:fe
3       RSUDAM2                           active              8c:b6:4f:e9:f3:42
4       RNORM6                            active              10:8c:cf:1d:fd:42
```

It gives the MAC address of individual VDC belong to the Nexus 7k Switch. This is used to update default VDC *ComposedOf* relation with the other VDCs that are created.

Note If the default VDC goes down, then the other VDCs are also impacted. However, IP 9.1 generates only one **Down** root cause instead of four individual VDC Down problems.

Identifying AggregationPorts through CLI

To identify AggregatePorts and figure out ports that are part of an aggregation, run the command in CLI:

```
show port-channel summary
```

Sample output

```
lima1054# show port-channel summary
Flags:  D - Down        P - Up in port-channel (members)
        I - Individual  H - Hot-standby (LACP only)
        s - Suspended   r - Module-removed
        S - Switched    R - Routed
        U - Up (port-channel)
```

```
--------------------------------------------------------------------------
Group Port-       Type     Protocol  Member Ports
      Channel
--------------------------------------------------------------------------
1     Po1(SU)     Eth      LACP      Eth1/19(P)   Eth1/20(P)
```

Initiating discovery

Table 11-5. Initiating manual discovery and autodiscovery

Initiating manual discovery	Initiating autodiscovery
The IP Manager provides two options for initiating a manual discovery, which are done through the Domain Manager Administration Console: ■ Option 1: Import a seed file that lists one or more seed systems to be added to the topology. This option is best for adding a set of systems. See Using the import from seed file command to initiate discovery ■ Option 2: Add seed systems to the topology one at a time by using the **Add Agent** command. This option is best for adding a small number of systems.See:Adding an SNMPv1/v2c seed system Adding an SNMPv3 seed system	When autodiscovery is enabled, use either option to initiate the autodiscovery process. When enabled and initiated, autodiscovery will occur whenever either a full discovery occurs or a system on the Pending Devices list is successfully discovered. Use any one of these two options: ■ Using the import from seed file command to initiate discovery, or ■ Adding an SNMPv1/v2c seed system Adding an SNMPv3 seed system

Initiating manual discovery and autodiscovery lists the options to initiate manual discovery and autodiscovery.

Using the import from seed file command to initiate discovery

A seed file consists of one or more seed systems, each of which represents a discovery candidate system. How to create a seed file provides instructions on creating a seed file.

To import a list of seed systems from a seed file:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   “Opening the Domain Manager Administration Console” on page 18 provides instructions on opening this console.

2 In the Domain Manager Administration Console, select **Topology** > **Import from seed file**, or click the **Import from seed file** toolbar button, to launch the **Import From Seed File** dialog box.

3 In the dialog box, specify the complete path and name of the seed file.

   By default, the path to the seed file is BASEDIR/smarts/local/conf. Note that the seed file must be stored on the host where the IP Manager runs.

4 Click **OK** to import the seed systems and to start the initial full discovery.

A Discovery Progress window opens, shows progress messages, and might show a list of systems (pending systems) that could not be discovered. When you see “Last discovery completed” near the end of the progress report in the Discovery Status section, the probing of topology information is complete.

Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides information about discovery errors, their solutions, and pending systems.

5 Click **Close** to close the Discovery Progress window.

Adding an SNMPv1/v2c seed system

To add an SNMPv1 or v2c seed system:

1 Attach the Domain Manager Administration Console to the target IP Manager.

“Opening the Domain Manager Administration Console” on page 18 provides instructions on opening this console.

2 In the Domain Manager Administration Console, select **Topology** > **Add Agent**, or click the **Add Agent** toolbar button, to display the **Add Agent** dialog box.

3 In the dialog box, specify the hostname or IP address of the v1/v2c seed system in the **Agent Name** field.

You can also specify the port on which the SNMP agent of the seed system receives requests by typing a colon (:) and the port number after the hostname or IP address. If you do not specify a port, the default port that is specified in the discovery.conf file is used.

- Optional: If the read community string of the v1/v2c seed system is not one of the strings in the **Read Community Strings** window (**Topology** tab) of the Domain Manager Administration Console, type a read community string in the **Community String** field. Adding read community strings provides more information about read community strings.
  - Optional: Select the **Advanced Options** checkbox to display the **Add Agent Advanced Options** panel. In the **Advanced Options** panel, you can optionally specify the SNMP port, the SNMP version, and the access mode for the v1/v2c seed system.

    Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides information about seed file options, formats and their values.
  - Click **OK** to add the v1/v2c seed system and to start the discovery process. The **Discovery Progress** window appears.

    Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides information about discovery errors and their solutions, and about pending systems.

Adding an SNMPv3 seed system

To add an SNMPv3 seed system:

1 Attach the Domain Manager Administration Console to the target IP Manager.

“Opening the Domain Manager Administration Console” on page 18 provides instructions on opening this console.

2 In the Domain Manager Administration Console, select **Topology** > **Add Agent**, or click the **Add Agent** toolbar button to display the **Add Agent** dialog box.

3 In the dialog box, specify the hostname or IP address of the v3 seed system in the **Agent Name** field.

You can also specify the port on which the SNMP agent of the seed system receives requests by typing a colon (:) and the port number after the hostname or IP address. If you do not specify a port, the default port that is specified in the discovery.conf file is used.

4 Select the **Advanced Options** checkbox to display the **Add Agent Advanced Options** panel.

In the **Advanced Options** panel, you can optionally specify the SNMP port and the access mode for the v3 seed system. Chapter 2 in VMware Smart Assurance IP Manager Concepts Guide describes these options and their values.

5 In the **Advanced Options** panel, select SNMP version v3 to display the **Add Agent SNMPv3 Specifications** panel.

In the **SNMPv3 Specifications** panel, you must specify the username for the v3 seed system; you can optionally specify the SNMP engine ID, the authentication protocol, the privacy protocol, and the context name.

If you specify an authentication protocol (MD5, SHA, or NONE—NONE is the default), you must specify an authentication password; the password that you type is masked with asterisks (*) for security reasons. If you specify NONE for the authentication protocol or leave the field empty, no password is required.

If you specify a privacy protocol (DES, AES, or NONE—NONE is the default), you must specify a privacy password; the password that you type is masked with asterisks (*) for security reasons. If you specify NONE for the privacy protocol or leave the field empty, no password is required.

VMware Smart Assurance IP Manager Concepts Guide describes these options and their values.

6 Click **OK** to add the v3 seed system and to start the discovery process. The **Discovery Progress** window appears.

Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides information about discovery errors and their solutions, and about pending systems.

Preparing for manual discovery

Manual discovery can be used to discover SNMPv1, v2c, or v3 systems in IPv4 or IPv6 networks.

- Using Manual discovery without autodiscovery — By using manual discovery without autodiscovery, the IP Manager will discover just the seed systems. The seed systems are presented to the IP Manager as entries in a seed file, rather than being presented individually by using the **Add Agent** command. A comprehensive seed file can only be created if the network topology information is complete and accessible. Using the Add Agent command to initiate discovery provides information on the **Add Agent** command.
- Using Manual discovery with autodiscovery — In situations where the network topology information is incomplete, unavailable, or constantly changing, manual discovery is combined with autodiscovery. The seed file can be much smaller because the seed systems that are specified for manual discovery also serve as a starting point for autodiscovery. Autodiscovery automatically discovers the managed network from the seed systems.

  For manual discovery, or manual discovery that is combined with autodiscovery, you create a seed file by editing the BASEDIR/smarts/conf/seedfile template in the IP Management Suite installation area. How to create a seed file provides the procedure for creating seed files.

Using the Add Agent command to initiate discovery

The Add Agent command is used to enter individual seed systems. You can specify a seed system as a hostname or as an IP address:

- For a hostname, you can specify a hostname with or without an explicit IP version suffix (:v6v4, :v4v6, :v6, or :v4).

  Here is an example hostname entry that includes a v4 suffix: Core-ROUTER-1:v4. If you do not include an explicit IP version suffix, the SM_IP_VERSIONS environment variable controls how the local name resolution service resolves the hostname to a corresponding IP address. By default, SM_IP_VERSIONS is set to v6v4.
- For an IP address, you can specify an IPv6 or IPv4 address.

  If you specify an IPv6 address, enclose the address in square brackets; for example: [2001:0db8::0010].

  Here is an example IPv6 address entry that includes a port number: [2001:0db8::0010]:200.

  Note Do not add an overlapping IP address as a seed system.

  Chapter 8, IPv6 Address Conventions, in the VMware Smart Assurance IP Manager Reference Guide contains information on Hostname syntax. Setting SM_IP_VERSIONS describes the procedure to set this environment variable. Overlapping IP addresses are discussed in the VMware Smart Assurance IP Manager Concepts Guide.

Preparing for autodiscovery

Autodiscovery can be used to discover SNMPv1 or v2c systems in IPv4 networks. Autodiscovery requires discovery filters. For systems that match a discovery filter and are added to the modeled topology, they too are probed for IP addresses of their neighbors. The autodiscovery cycle continues until no more new IP addresses match the discovery filters. The discovery filters, which are configurable, ensure that the IP Manager sweeps only the networks that the customer systems are configured to access. Preparing for autodiscovery consists of the following tasks as listed in Tasks to prepare for autodiscovery:

Table 11-6. Tasks to prepare for autodiscovery

Tasks	Type	Purpose	
Create seed file to seed the topology or Use **Add Agent** command	Mandatory	Autodiscovery uses the IP addresses and topology that are gathered from seed systems to discover all of the systems in the managed network. The IP addresses or system names of the seed systems are specified in a seed file or in **Add Agent** commands.	■ How to create a seed file
Create discovery filters	Mandatory	For autodiscovery, discovery filters determine which systems are discovered and added to the modeled topology. You must create at least one discovery filter for autodiscovery.	■ *"Starting discovery" on page 87* ■ How to create a discovery filter ■ How to delete a discovery filter
Specify autodiscovery safeguards	Optional	To ensure that autodiscovery does not automatically add to the modeled topology to those systems that you do not intend to manage, use one or both of the following safeguards:	■ "If you use cron or sm_sched or some other scheduler utility to schedule pending discoveries, set the Interval field for pending discovery to 99 days in the Domain Manager Administration Console." on page 87 ■ Setting a current system limit for the topology
Select autodiscovery data source	Optional	The Name Resolution, IP Network, and Neighbor probes contact the autodiscovery data sources to gather the topology and IP address table (ipAddrTable) MIBs of the seed systems to determine what other systems are known to the seed systems.	■ Procedure to select autodiscovery data sources
Enable autodiscovery	Mandatory		■ Enable autodiscovery

In the discussions, you will see several references to the Pending Devices list, Discovery Progress window, pending discovery, and full discovery for the IP Manager.

Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides information about the Pending Devices list that appears in the Discovery Progress window, and Scheduling automatic discovery provides information about pending discovery and full discovery.

SNMP v3 auto discovery

IP Manager has been enhanced to support auto discovery of devices in an IP network using SNMP v3 polling mechanism. With the support for SNMP v3 polling mechanism, you can discover and monitor those devices in the network that supports SNMP v3.

This has been facilitated by the following changes in the IP Domain Manager Administration Console user interface:

Enhancement to Discovery Filters tab

Enhancement to Topology tab

Supported Encryption Standards for SNMP v3 autodiscovery

Enhancement to Discovery Filters tab

The Discovery Filters tab has been enhanced with a section on SNMP v3 Specifications. This section is enabled when the option V3 or AUTODETECT is selected in Extended Filter Information >SNMP Version drop-down list.

Note For SNMP v3 autodiscovery, it is recommended to (create the discovery filters and) provide the SNMP v3 credentials in the Discovery Filters tab instead of in the Topology tab. This is because the (filters and) credentials provided in the Discovery Filters tab are applied only to the specific IP address range, thereby reducing the discovery time.

For SNMP v3 autodiscovery, atleast one default SNMP v2c community string should be provided, either in the **Topology**tab > **Read Community Strings** field or in the discovery.conf file. The read community string is a mandatory input for SNMP v3 autodiscovery, without which SNMP V3 autodiscovery will fail. This also applies when the option **V3**is selected in **SNMP Version**field in the Discovery Filters tab.

1 In Domain Manager Administration Console >Discovery Filters tab, select **View Extended Filter**option.

2 In the **Extended Filter Information**section, select any of the following options in the the **SNMP Version**drop-down list:

- AUTODETECT. When you select this option, it is mandatory to:
  - Set the fallback options in Topology tab > **AutoDiscovery AUTODETECT Priority Options**drop-down list.
    - Create the SNMP v3 credentials, either in the Discovery Filters tab or in the Topology tab. To create SNMP v3 credentials, see step 3 on page 83 .
  - V1
  - V2C

- V3

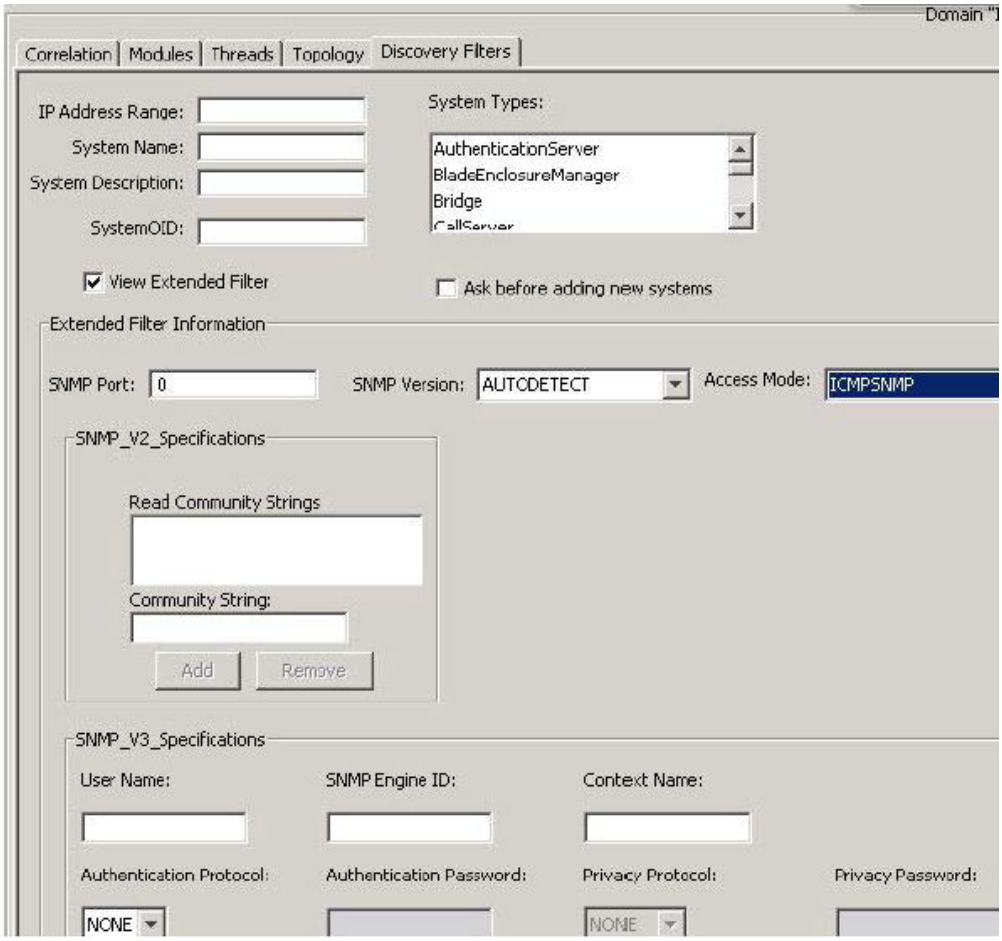
Note If you select V3, then it is required to provide the SNMP v3 credentials of the devices either in the Discovery Filters tab or in the Topology tab. See step 3 on page 83 . If you do not provide SNMP v3 credentials in the Discovery Filters tab, then it will pick up the SNMP v3 credentials provided in the Topology tab.

3 To create SNMP v3 credentials, provide the following details in the **SNMP_V3_Specification** section as shown in Discovery Filters tab:

- **User Name**— Should comprise of 32 characters at the most.
  - **SNMP Engine ID**— The characters count should be an even number, comprising of atleast 10 characters.
  - Context name
  - **Authentication Protocol**. Select any one of the following options:
    - None
    - MD5 (MD5 is not supported in FIPS mode)
    - SHA
  - Authentication Password
  - **Privacy Protocol.**Select any one of the following options:
    - None
    - DES (DES is not supported in FIPS mode)
    - 3DES (3DES is not supported in FIPS mode)
    - AES (AES refers to AES128).
    - AES 192
    - AES256

- Privacy Password

Figure 11-1. Discovery Filters tab



4 Click **Add Credentials**. The credential information gets added and they appear in the SNMP V3 Credentials section just below. You can also modify and delete the credentials.

Note In the discovery.conf file, a new variable “MaximumSNMPV3Credentials” is added which has the default value of 4.

When triggering SNMP v3 autodiscovery, if the option “Ask before adding new systems” is selected, then the new devices will not be discovered using SNMP v3 discovery. IP Manager will try to discover the new devices using SNMP v2. If SNMP v2 is not configured on the real device, discovery will fail. However, if the “Ask before adding new systems” option is not selected, then SNMP v3 autodiscovery works without any errors.

Enhancement to Topology tab

The SNMP v3 credentials created in the Topology tab get applied to all discovery filters in the Discovery Filters tab, thereby increasing the discovery time.

The Topology tab has been enhanced with the **Autodiscovery AUTODETECT Priority Options** drop-down list. You can select any one of the following fallback options for SNMP polling mechanism:

- V3V2V1
- V2V1V3

  Note The fallback options need to be selected based on the number of V3 or V2 devices in the network. For example, if there are more number of V3 devices, then select V3V2V1, else select V2V1V3.

  For the options V3V2V1 and V2V1V3, ensure that SNMPv3 credentials of the devices are present either in the Discovery Filters tab or in the Topology tab. Else, SNMP v3 discovery will fail.

- V2V1 - This is the default option.

  Note In the Discovery Filter tab, if you select the AUTODETECT option in the SNMP version field, then it is mandatory to set the **Autodiscovery AUTODETECT Priority Options** in the Topology tab.

Supported Encryption Standards for SNMP v3 autodiscovery

The SNMP v3 autodiscovery feature in this release supports the following encryption standards:

- Advanced Encryption Standard (AES128)
- Advanced Encryption Standard (AES256)
- Advanced Encryption Standard (AES192)
- Triple Data Encryption Standard (Triple DES)

  Note IP Manager follows the convention where AES means AES128.

Scheduling automatic discovery

You can schedule full discovery and pending discovery intervals:

- Scheduling a full discovery interval determines how often the IP Manager attempts to rediscover the existing systems in its repository.
- Scheduling a pending discovery interval determines how often the IP Manager attempts to discover the systems on the Pending Devices list.

  By default, the full discovery interval is enabled and set to 1 day, and the pending discovery interval is enabled and set to 6 hours. The parameters that control scheduled discovery are located on the Topology tab page of the Domain Manager Administration Console.

The full discovery interval should not be set to less than 60 minutes, and the pending discovery interval should not be set to less than 5 minutes. Violating either of these restrictions will result in the printing of an exception by the Domain Manager Administration Console. For example:

```
Apply changes failed: Invalid full discovery interval
```

Configure the full discovery interval

The full discovery interval is counted from the time when the IP Manager is started. So, if the full discovery interval is set to 1 day and the IP Manager is started at 5 A.M., the full discovery will occur each day at 5 A.M.

The ICF-TopologyManager program for the IP Manager starts a full discovery every full discovery interval unless the previous full discovery is still in progress, in which case the ICF-TopologyManager ignores the discovery request and writes an exception to the IP Manager's log file. For example, if the full discovery interval is set to 1 hour and the actual full discovery takes 1.3 hours to complete, the ICF-TopologyManager will ignore the discovery request every other discovery interval, which means that the actual full discovery will occur every 2 hours.

To specify a scheduled full discovery interval other than 1 day:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   *"Opening the Domain Manager Administration Console" on page 18*provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, click the IP Manager name to display a multiple-tab window in the right panel of the console.

3 Click **Topology**.

4 Type a value in the **Interval** field for full discovery and select a time interval (**Seconds**, **Minutes**, **Hours**, or **Days**) from the list box. Ensure that the interval is equal to or greater than 60 minutes.

5 Click **Apply**.

   As an alternative to using the Domain Manager Administration Console to specify a full discovery interval, you can use the **cron** or **sm_sched** scheduler and the **dmctl** or **sm_tpmgr** command to schedule a full discovery at a specific time. Examples of **dmctl** and **sm_tpmgr** commands to initiate a full discovery are given in *"Starting discovery from the command line" on page 88*.

   If you use **cron** or **sm_sched** or some other scheduler utility to schedule full discoveries, clear the *Enable Full Discovery* checkbox in the Domain Manager Administration Console.

Configure the pending discovery interval

The pending discovery interval may be configured to a different value than the full discovery interval.

The pending discovery interval is counted from the time when the IP Manager is started. So, if the pending discovery interval is set to 6 hours (the default) and the IP Manager is started at 5 A.M., the pending discovery will occur every 6 hours thereafter, that is, at 11 A.M., at 5 P.M., at 11 P.M., and so on.

To specify a pending discovery interval other than 6 hours:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   *“Opening the Domain Manager Administration Console” on page 18*provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, click the IP Manager name to display a multiple-tab window in the right panel of the console.

3 Click **Topology**.

4 Type a value in the **Interval** field for pending discovery and select a time interval (**Seconds**, **Minutes**, **Hours**, or **Days**) from the list box. Ensure that the interval is equal to or greater than 5 minutes.

5 Click **Apply**.

   Like full discovery, you can use the **cron** or **sm_sched** scheduler and the **dmctl** or **sm_tpmgr** command to schedule a pending discovery at a specific time. Examples of **dmctl** and **sm_tpmgr** commands to initiate a pending discovery are given in Starting pending discovery from the command line.

   If you use **cron** or **sm_sched** or some other scheduler utility to schedule pending discoveries, set the Interval field for pending discovery to 99 days in the Domain Manager Administration Console.

Starting discovery

You can start the discovery process in any one of the following ways:

- *“Starting discovery from the console” on page 87*
- *“Starting discovery from the command line” on page 88*

Starting discovery from the console

From a Domain Manager Administration Console that is attached to the IP Manager, you can use any of the following methods to manually invoke a discovery:

- To discover all managed systems, including those on the Pending Devices list, select **Topology** > **Discover All**.
- To discover all systems on the Pending Devices list, select
- **Topology** > **Discover Pending**.
- To discover a single system on the Pending Devices list, right-click the system on the Pending Devices list and then select **Rediscover** from the pop-up menu.
- To rediscover a single system in the topology tree, click the system and then select **Topology** > **Rediscover**.

  You can also rediscover a system by right-clicking it and selecting **Rediscover** from the pop-up menu.

Starting discovery from the command line

From the command line, you can invoke a full discovery or a pending discovery by using the **dmctl** or **sm_tpmgr** utility. Doing so is useful if you want to schedule a full or pending discovery at a specific time by using batch processing or the **sm_sched** utility.

The **dmctl** and **sm_sched** utilities are described in the HTML pages that are located in the BASEDIR/smarts/doc/html/usage directory of any Domain Manager installation area.

Starting full discovery from the command line

To use the **dmctl** utility to initiate a full discovery, go to the BASEDIR/smarts/bin directory in the IP Manager installation area and invoke the following command on one line:

```
dmctl -s <
   IP Manager

 instance name>
-b <
brokerName:port
>
invoke

  ICF_TopologyManager::ICF-TopologyManager discoverAll
```

For example:

```
dmctl -s INCHARGE-AM -b localhost:426 invoke

  ICF_TopologyManager::ICF-TopologyManager discoverAll
```

When prompted for a username and password, specify an administration account; for example, admin and changeme.

To use the **sm_tpmgr** utility to initiate a full discovery, go to the BASEDIR/smarts/bin directory in the IP Manager installation area and invoke the following command:

```
sm_tpmgr -s <
   IP Manager

 instance name>
-b <
brokerName:port
> --discover-all
```

For example:

```
sm_tpmgr -s INCHARGE-AM -b localhost:426 --discover-all
```

Starting pending discovery from the command line

To use the **dmctl** utility to initiate a pending discovery, go to the BASEDIR/smarts/bin directory in the IP Manager installation area and invoke the following command on one line:

```
dmctl -s <
   IP Manager

 instance name> -b <
brokerName:port
> invoke

  ICF_TopologyManager::ICF-TopologyManager discoverPending
```

When prompted for a username and password, specify an administration account; for example, admin and changeme.

To use the **sm_tpmgr** utility to initiate a pending discovery, go to the BASEDIR/smarts/bin directory in the IP Manager installation area and invoke the following command:

```
sm_tpmgr -s <
   IP Manager

 instance name> -b <
brokerName:port
> --discover-pending
```

Creating discovery filters

This section lists all the tasks related to the creation of discovery filters. Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide contains more information about discovery filters.

How to create a discovery filter

For autodiscovery, discovery filters determine which systems are discovered and added to the modeled topology. You create a filter by specifying a combination of characters and wildcards, and by selecting the types of systems to which the filter is applied.

Chapter 9, Wildcard Patterns, in the VMware Smart Assurance IP Manager Reference Guide provides information on wildcard syntax.

To create a discovery filter:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   *“Opening the Domain Manager Administration Console” on page 18*provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, click the IP Manager name to display a multiple-tab window in the right panel of the console.

3 Click **Discovery Filters**.

4 Specify values for each of the discovery filter fields:

   - IP Address Range
     - System Name
     - System Description
     - SystemOID
     - System Types

       You must specify a value for each field. The System Name and System Description fields are case-sensitive.

5 Decide whether newly discovered systems that match this filter should be added to the topology or to the Pending Devices list.

   Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides information about automatically or manually accepting new systems.

6 Optional: Select **View Extended Filter** to view the extended options of the filter. The extended options enable you to specify the SNMP port, the SNMP version, the access mode, and the read community string or strings to be used when polling the systems that match an IP address within the IP address range of this filter.

7 Click **Add**, and then click **Apply**.

   When you create multiple filters, a candidate system is compared against the filters in the order in which the filters are arranged. When a candidate system matches a filter, the system is added to the discovery queue. Because of this behavior, put the more limiting filters first. You can arrange the order of the filters by selecting a filter and clicking the up or down arrows under the list of filters.

How to delete a discovery filter

To delete a discovery filter:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   “Opening the Domain Manager Administration Console” on page 18 provides instructions on opening this console.

2 In the left panel of the **Domain Manager Administration Console,** click the **IP Manager** name to display a multiple-tab window in the right panel of the console.

3 Click **Discovery Filters**.

4 Select the discovery filter that you want to delete.

5 Click **Delete**, and then click **Apply**.

Using manual accept mode

You can configure a discovery filter for manual accept mode so that a candidate system that matches the filter is added to the Pending Devices list in the THROTTLED state. By default, a candidate system that matches a filter is automatically discovered and added to the modeled topology.

To enable manual accept mode for a filter:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   "Opening the Domain Manager Administration Console" on page 18 provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, select the IP Manager name to display a multiple-tab window in the right panel of the console.

3 Click **Discovery Filters**.

4 Select the filter for which you want to enable the manual accept mode.

5 Select **Ask before adding new systems** to enable manual accept mode for the filter.

6 Click **Apply**.

   After autodiscovery is enabled and initiated, you can view the Pending Devices list to see which systems passed the filter and are waiting to be accepted. Systems that passed the filter and are waiting to be accepted are marked as THROTTLED. Accepting a THROTTLED system changes the state of the system to UNDISCOVERED. An UNDISCOVERED system will be discovered during the next pending discovery interval.

   Note Right-click a system and select the Rediscover option to discover the system before the next pending discovery takes place.

Setting a current system limit for the topology

You can limit the number of systems that are added to the modeled topology by setting a current system limit. By default, the current system limit is set to 50 systems.

To change the current system limit:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   "Opening the Domain Manager Administration Console" on page 18 provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, click the IP Manager name to display a multiple-tab window in the right panel of the console.

3 Click **Topology**.

4 In the **Current System Limit** field, type the total number of systems that you want to allow into the modeled topology.

5 Click **Apply**.

Note The current system limit applies to both manual discovery and autodiscovery.

When the current system limit is exceeded, the discovery process will complete the discovery of any system (and its components) in the midst of being discovered, including all virtual routers that are configured for a router or switch that implements virtual routers. When a router or switch that is implementing virtual routers is discovered, the chassis-based router/switch is counted as a system, and each virtual router is counted as a system. Any new discovery candidate systems will be added to the Pending Devices list in the UNDISCOVERED state, and the message “System limit exceeded” will appear in the Discovery Status section of the Discovery Progress window.

Procedure to select autodiscovery data sources

To select which data sources are to be used for autodiscovery:

1 Attach the Domain Manager Administration Console to the target IP Manager.

“Opening the Domain Manager Administration Console” on page 18 provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, click the IP Manager name to display a multiple-tab window in the right panel of the console.

3 Click **Topology**.

4 Select the data sources that are to be used by autodiscovery to find additional candidate systems. The data sources are listed under **Enable Auto Discovery**.

5 Click **Apply**.

Enable autodiscovery

To enable autodiscovery:

1 Attach the Domain Manager Administration Console to the target IP Manager.

“Opening the Domain Manager Administration Console” on page 18 provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, click the IP Manager Domain Manager name to display a multiple-tab window in the right panel of the console.

3 Click **Topology**.

4 Select **Enable Auto Discovery**.

5 In the **Community String** field, provide any additional read community strings that are needed by the IP Manager to be able to communicate with systems on your network. You may add up to four read community strings; the default is *public*.

Note For Cisco devices, do not configure the community string with a “@” character.

The community strings that are specified here are global default community strings that apply to autodiscovery and manual discovery. Adding read community strings provides the procedure for adding community strings.

6 Click **Apply**.

OID Info Tool

The OID Info tool enables you to view the list of OIDs that are used to discover and monitor devices such as Host, Router, Switch, and so on. The OID information helps identify faults in the configuration of the SNMP agent of the device. The OID Info tool displays the list of OIDs and their SNMP values for a device. It fetches the value of the OIDs from the device and compares the SNMP values against the values in the topology.

The OID Info tool is introduced in the IP Manager and SAM Console versions 9.2 and is not available in older versions. The OIDs can be viewed through the command line or through the **OID Info** option in the Domain Manager Administration Console.

The **OID Info** option will not be enabled if:

- The system object is discovered in ICMP only mode.
- The device does not have a SysOID.
- The device does not have an associated SNMPAgent object in the topology.
- An older version of IP Manager is attached to the Domain Manager Administration Console.

The OIDs are classified into two types – Discovery and Monitoring. In case of Discovery type, only statically configured OIDs are displayed. OIDs that the IP domain manager queries dynamically will not be displayed. The tool displays the name of the discovery probe and the associated discovery driver.

In case of Monitoring type, the name of the instrumentation class and the device component associated with it is displayed.

The VMware Smart Assurance IP Manager Concepts Guide and the VMware Smart Assurance IP Manager Reference Guide provide more information on the OID Info tool.

Configure OID Info tool

You must configure the **oidInfo.conf** file to view the OID information of a device through the command line interface. The file is located under <BASEDIR>/conf/discovery.

To edit the **oidInfo.conf** file, go to the <BASEDIR>/conf/discovery folder of your installation and type the following command:

```
./sm_edit conf/discovery/oidInfo.conf file
```

You must specify the following information for each device in the oidInfo.conf file:

- **Class** – The system class such as Router, Switch, Host, and so on.
- **Instance** – The name of the instance of the system class specified. The value of this should be the same as the value of the Name attribute of the system object in the IP topology. Other attribute values such as the DisplayName or the IP Address of the system object are not valid.
- **Type** – The type of class. You can enter one of the following values:
  - Discovery –To view the OIDs of classes that are used for discovery only
  - Monitoring – To view the OIDs of classes that are used for monitoring only
  - All – To view the OIDs of classes that are used for both discovery and monitoring

    The value for this parameter is case sensitive and will assume a default value of *Monitoring* if values apart from those specified above are used.
- **Entity** – This parameter is applicable when the value of the*Type* parameter is **Monitoring**. The value of this parameter will be ignored if the *Type* is **Discovery**. By default the value of this parameter is “*”. This parameter is used to filter out device components that will be displayed in the output. The values will need to have the class names of the component entities of a device such as Card, Interface, Port, Fan, Memory, and so on.

  For example,
  - To view the OIDs related to monitoring of Ports, Interfaces, and Memory, the pattern can be specified as **Port|Interface|Memory.**
  - To view the OIDs for all entities except Ports and Interfaces, the pattern can be specified as **~Port|~Interface**.

    Additionally, you can also specify the polling interval and the SNMP timeout for the following parameters:
- pollingInterval – The interval in seconds at which values need to be polled for all the OIDs. The default value is 120.
- snmpTimeout – The timeout in seconds for SNMP requests from the tool. The default value is 500.

  The output for each entry configured in the oidInfo.conf is saved in a separate text file whose name is of the format <ClassName_InstanceName.txt>. The file is located at <BASEDIR>/logs/oidInfoLogs. If the file already exists, it will be renamed and a timestamp will be appended to the filename.

Displaying OIDs in the command line

Go to the <BASEDIR>/smarts/bin folder of IP Manager and type the following command to run the sm_oidInfo.pl script:

```
./sm_perl sm_oidInfo.pl <options>
```

The output from running the OID Info tool is saved in the sm_oidInfo.log file located at <BASEDIR>/logs/oidInfoLogs. A log file is generated everytime you run this tool. If a log file already exists from the previous run, it will be renamed and a timestamp will be appended to the file name.

OID Info tool allows you to view the OID information of devices in other installations of IP Manager from your current installation.

The following procedure can be performed if you want to view the OID information of devices discovered in a different installation of the IP Manager:

1 Edit the oidInfo.conf file of your current IP 9.2 installation at <BASEDIR>/conf/discovery to provide the information of devices discovered in a different installation.

2 Go to the <BASEDIR>/smarts/bin folder of your current IP 9.2 installation and type the following command:

```
./sm_perl sm_oidInfo.pl --server=<server_name> --broker=<host:port>
```

sm_oidInfo modes of operation

The sm_oidInfo has the following command-line options:

- --server – Specifies the name of the IP domain manager where the devices to be used for running the tool are discovered. This option is mandatory because the tool needs to connect to the server to retrieve the required configuration and OID data from the server.
- --broker – Specifies the broker at which the IP domain manager is registered. If not specified, the tool will connect to the broker defined by the SM_BROKER or SM_BROKER_DEFAULT environment variables. If a broker can not be determined, the tool will report an error and exit.
- --displayValues – To gather the SNMP values of the OIDs.
- --compareValues – To compare the OID's topology and SNMP values.
- --poll – To poll and display, or compare the OID values *n* times. The maximum value of *n* is 100.
- --silent
- --help

  When the tool is run by not specifying any <*options*> , the output will contain the list of OIDs for only those devices that are configured in the oidInfo.conf.

To display OIDs in the console

1 Attach the IP domain manager to the Domain Manager Administration Console.

2 Expand the tree on the left pane and navigate to the device.

3 Right-click the device and select **OID Info**.

The **OID Info** window displays the following columns:

- **Type** –The type of class, Discovery or Monitoring.
- **Component** – Contains the Probe Name such as Containment and Performance in case of Discovery type, and Entity name such as Interfaceand Port in case of Monitoring type.

  Note The Bridge-probe component is only available for Discovery type classes, Switch and Bridge.

- **Entity** – Contains the Driver Name in case of Discovery type and the Instrumentation class name in case of Monitoring type.
- **OID** – The SNMP OID associated with the device.
- **Description** – Description of the SNMP OID or discovery probe if available.
  - For Monitoring type, this field contains the attribute name of the SNMP OID or discovery probe.
  - For Discovery type, due to the limitations of IP Manager, this field is either empty or contains a generic description of the SNMP OID or discovery probe.

    Note Monitoring classes of devices that donot have an OID associated with it, will not be listed in the OID Info window.

    The **OID Info** option is disabled for the following object types because these objects do not have a system OID or an SNMP agent associated with them:

- AuthenticationServer
- BladeEnclosureManager
- VSSChassis
- WirelessClient

Note Objects discovered in ICMP only mode, objects not having a SysOID, objects not having an associated SNMPAgent object in the topology, and thin clients WirelessAccessPoints will also be ignored.

Seed file related tasks

This section provides the list of tasks related to the seed file:

- How to create a seed file
- How to encrypt passwords in the seed file
- Extracting a seed file from the IP Manager

How to create a seed file

Typically, you use the template file named seedfile, which is located in the BASEDIR/smarts/conf directory, as the basis for creating a seed file. You use the seedfile template to create a seed file for each IP Manager in your deployment.

If you start multiple IP Managers from the same IP Manager installation directory, use the seedfile template to create seed files with distinct names. For example, if you plan on starting two IP Manager instances named East and West from the same IP Manager installation directory, consider naming the two seed files East_seedfile and West_seedfile.

To use the seedfile template to create a seed file:

1. Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the seedfile template:

   ```
   sm_edit conf/seedfile
   ```

2. Add the seed entries for the systems to be discovered.
3. Save and close the file.

   The modified version of the seedfile template is saved to the BASEDIR/smarts/local/conf directory.

How to encrypt passwords in the seed file

When viewing the seed file with the **sm_edit** utility, add the following line (including #) as the first line in the seed file to direct **sm_edit** to encrypt the authentication and privacy passwords in the file:

```
#<encrypted seed>:1.0:AUTHPASS,PRIVPASS
```

When you close the seed file, **sm_edit** will automatically encrypt the authentication and privacy passwords in the file.

To determine whether a password in a seed file has been encrypted, look for the prefix <E-1.0> in the password. If <E-1.0> is present, the password is encrypted; if <E-1.0> is not present, the password is plain text. Although seed files that contain plain-text passwords are supported, they are not recommended.

The **sm_edit** utility does not encrypt a password that is already encrypted, which is determined by the presence of the <E-1.0> prefix. Once a password is encrypted, you cannot use **sm_edit** to decrypt the password. If you want to change an encrypted password, you must delete the password, including the <E-1.0> prefix, type the new password, and then save the file.

During the installation of the IP Manager, the installation program uses the case-sensitive secret phrase *Not a secret* to create a secret key, which is to be used by **sm_edit** and other VMware Smart Assurance utilities to encrypt passwords. The secret phrase and secret key, themselves, are encrypted and stored in the BASEDIR/local/conf/imk.dat file. After the installation, an VMware Smart Assurance administrator should use the **sm_rebond** utility to change the secret phrase and secret key.

The VMware Smart Assurance System Administration Guide provides information about **sm_rebond** and VMware Smart Assurance encryption.

Extracting a seed file from the IP Manager

You can retrieve a list of the IP addresses and read community strings of the SNMP agents that are managed by the IP Manager.

To obtain a list of SNMP agents, invoke the **sm_tpmgr** utility on the host where the IP Manager is running. This utility produces a text file named seedfile.log that lists the IP address of the SNMP agents in one column, and their corresponding read community strings in a second column. For SNMPv3 agents, of course, no read community strings appear in the second column.

To obtain a list of SNMP agents and their read community strings, type the following command from the BASEDIR/smarts/bin directory:

```
sm_tpmgr -s INCHARGE-AM --dump-agents --output=seedfile
```

where:

- The **-s** option specifies the name of the IP Manager.
- The **-- dump-agents** option extracts the information about the SNMP agents from the IP Manager.
- The **-- output=seedfile** option creates a file named seedfile.log in the BASEDIR/smarts/local/logs directory.
- Use the **-- help** option to learn more about the **sm_tpmgr** utility.

Discovery and short discovery of IPSec tunnels

The IP Manager discovers IPSec tunnels and Internet Key Exchange (IKE) tunnels between IPSec-enabled routers on IPv4 networks. It uses Router, Interface, and IP object types to model the tunnels.

During a full discovery, the IP Manager reads the Routing Table of IPSec-enabled routers to discover the terminating subinterfaces (WAN endpoints) of IPSec tunnels and IKE tunnels. The IP Manager layers the subinterfaces over the underlying physical interfaces and creates the network connections between the local and remote subinterfaces to form the IPSec and IKE tunnels.

By default, the IP Manager is not enabled to discover IPSec tunnels. To enable IPSec tunnel discovery, set the EnableIPSecDiscovery parameter in the tpmgr-param.conf file and the autoReprobe_short parameter in the discovery.conf file to TRUE. Also set SM_LOGICALCONNECTIONDOWN to DISABLED in the runcmd_env.sh file.

Short discovery configurations

By default, short discovery is not enabled. To enable short discovery, set the autoReprobe_short parameter in the discovery.conf file to TRUE:

- When autoReprobe_short is TRUE and EnableIPSecDiscovery in the tpmgr-param.conf file is TRUE, short discovery accepts both IsDownOrFlapping interface attribute changes and OperationallyDown card events as input.
- When autoReprobe_short is TRUE and EnableIPSecDiscovery is FALSE, short discovery accepts only OperationallyDown card events as input.

  To change the short discovery interval or the number of short discovery threads, modify the reprobePeriod_short parameter or the numberShortProbeThreads parameter in the discovery.conf file.

Analyzing IPSec tunnels

As a precondition for IPSec tunnel analysis, a user must add the following line to the BASEDIR/smarts/conf/local runcmd_env.sh file:

```
SM_LOGICALCONNECTIONDOWN=DISABLED
```

The Interface::Down problem, NetworkConnection::Down problem, and the DownOrFlapping impact events are described in the VMware Smart Assurance IP Manager Reference Guide.

Customizing discovery

This sections contains procedures that you can use to customize discovery.

Adding device entries to oid2type_Field.conf

To add device entries to the oid2type_Field.conf file:

1. Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the oid2type_Field.conf file:

   ```
   sm_edit conf/discovery/oid2type_Field.conf
   ```

2 Observe the syntax of device entry below and add the devices that you want to certify. Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides more information on syntax of device entries in oid2type_Field.conf.

```
# One or more commented-out lines that identify the device
<sysObjID> {
TYPE = <unitary computer system class>
VENDOR = <manufacturer>
MODEL = <model name>
CERTIFICATION = TEMPLATE
<discovery driver keyword> = <driver parameter name>
.
.
.
<discovery driver keyword> = <driver parameter name>
INSTRUMENTATION:
<instrumentation keyword> = <instrumentation set>
.
.
.
<instrumentation keyword> = <instrumentation set>
}
```

3 Save and close the file.

The modified version of the oid2type_Field.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

4 Restart the IP Managers to implement your changes.

Or, as an alternative to restarting the IP Managers:

For each IP Manager in your deployment, issue the following command from BASEDIR/smarts/bin to implement your changes:

```
sm_tpmgr -s <
   IP Manager

 instance name> -b <
brokerName:port
> --reloadoid
```

Modifying discovery hook scripts

You use the **sm_edit** utility to edit a discovery hook script to add the ASL convenience patterns, the parsing rules, and the processing rules that perform the actual processing. You can also add additional initialization code, or additional EOF or DEFAULT code, or both.

To modify a discovery hook script:

1 Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the script. For example:

```
sm_edit rules/discovery/custom/custom-end-system.asl
```

2 Add your ASL code to the script.

3 Save and close the script.

The modified version of the script is saved to the BASEDIR/smarts/local/rules/discovery/custom directory.

4 Restart the IP Manager to implement your changes.

Included in the IP Manager software distribution is an example of a modified custom-end-system.asl script. That script is located in the following directory:

BASEDIR/smarts/examples/dynamic-model/example2/rules/discovery/custom

Polling And Threshold

12

This chapter includes the following topics:

- Overview of Polling and Thresholds Console
- Opening the Polling and Thresholds Console
- Working with groups and settings
- Performing tasks in the Polling tab
- Performing tasks in the Thresholds tab
- External Polling Support for FibreChannelSwitchPort
- Policy Settings to control the managed state of interfaces
- Performing tasks in the Device Access tab
- Performing tasks in the Tagging tab

Overview of Polling and Thresholds Console

The Polling and Thresholds Console is used to configure groups and settings. You can:

- Define polling interval for different groups of network devices
- Set threshold values for monitoring those devices
- Specify login credentials
- Define IP Tagging

Opening the Polling and Thresholds Console

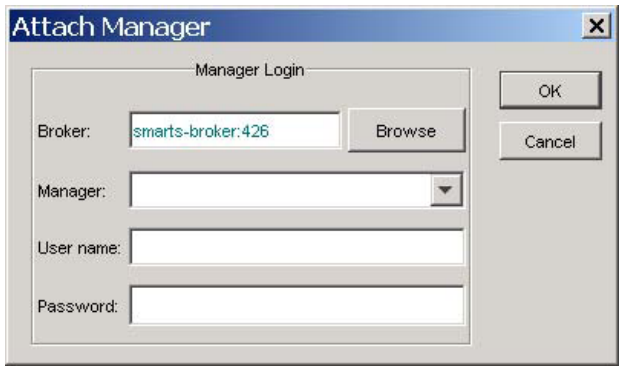
The Polling and Thresholds Console is used to display groups and modify their properties. To open the Polling and Thresholds Console:

1 Start the Global Console.

- On a UNIX system, go to the *BASEDIR/smarts/bin* directory in the VMware Smart Assurance Service Assurance Manager (Global Manager) installation area and enter **sm_gui**.

  The **Attach Manager** dialog box opens, an example of which is shown in Attach Manager dialog box.

  Figure 12-1. Attach Manager dialog box



2 In the dialog box,

- Ensure that the VMware Smart Assurance Broker for your deployment appears in the **Broker** text box.
  - Click the **Manager** drop-down list or the **Browse** button to display a list of active (running) Managers, and from that list select an IP Manager application (for example, INCHARGE-AM) in your deployment as the Manager to which you want to connect.
  - Enter your login username and password. (The default administration login is username *admin* and password *changeme*.)
  - Click **OK**.

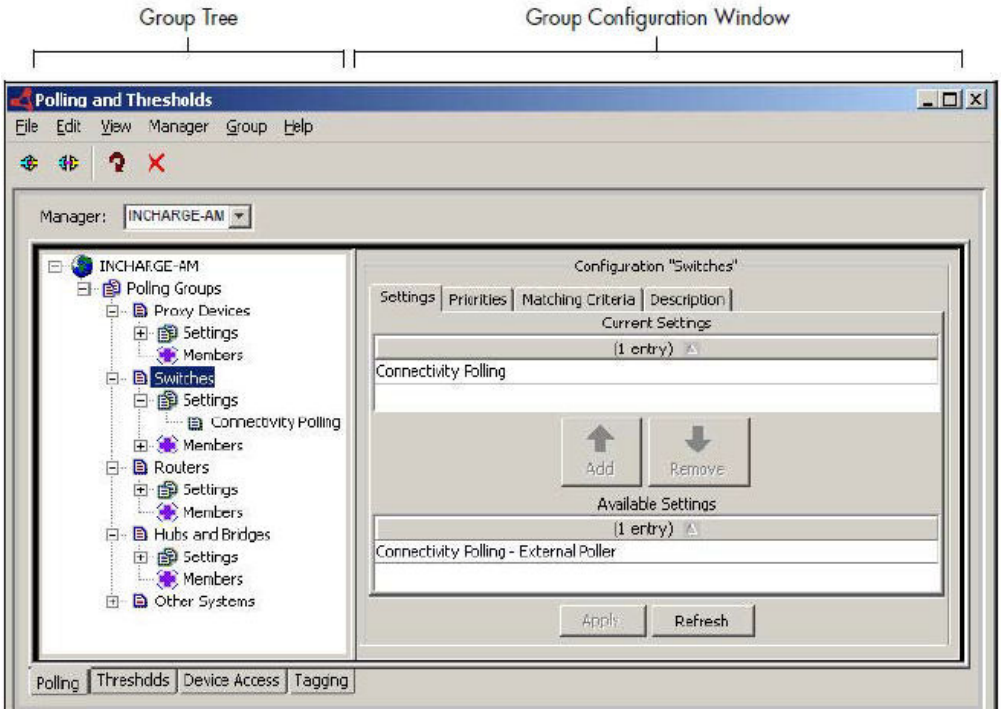
    The **Topology Browser Console** opens.

3 In the **Topology Browser Console**, choose **Configure > Domain Manager Administration Console**. The **Domain Manager Administration Console** opens.

4 In the **Domain Manager Administration Console**, choose **Configure > Polling and Thresholds**.

The **Polling and Thresholds Console** opens, an example of which is shown in Polling tab for IP Availability Manager.

Figure 12-2. Polling tab for IP Availability Manager



Layout of the Polling and Thresholds Console

When attached to an IP Availability Manager or IP Performance Manager, the Polling and Thresholds Console contains four tabs on the left pane, as shown in Layout of the Polling and Thresholds Console:

- Polling — Enables administrators to control the polling of the managed objects for monitoring purposes.
- **Thresholds —** Enables administrators to set thresholds for the polled data.
- **Device Access —** Enables administrators to specify credentials (such as login credentials) for device access.
- **Tagging —** Enables administrators to create IP tagging filters.

  The Polling and Thresholds Console is divided into two panels, as shown in Layout of the Polling and Thresholds Console:

- The **left panel** displays the icon for the Domain Manager in the upper-left corner and provides multiple tabs for configuration, depending on the Domain Manager to which the Polling and Thresholds Console is attached. Clicking a tab shows what groups, settings, and members are available on the tab.

  The **right panel** remains blank until a group, setting, or member is selected in the left panel. When an item is selected in the left panel, the right panel displays additional information regarding that item.

Polling and Thresholds Console toolbar buttons

The toolbar of the Polling and Thresholds Console provides quick access to the commands described in Polling and Thresholds Console toolbar buttons.

Table 12-1. Polling and Thresholds Console toolbar buttons

Button	Description
	Attach to a Manager
	Detach from a Manager
	Reconfigure a group
	Delete the selected item

Working with groups and settings

Using the Polling and Thresholds Console, you can perform the following tasks as listed in Tasks in polling and thresholds console:

Table 12-2. Tasks in polling and thresholds console

Tasks	Reference
Create new groups	Creating new groups
Copy existing group	Copying an existing group
Create empty groups	Creating an empty group
Modify properties of a group	Modifying the properties of a group
Add or remove settings from a group	Adding or removing Settings parameters
Modify priorities of a group	Modifying the priority of groups
Edit parameters of a setting	Modifying the values of Settings parameter
Restore default values of a setting	Restoring the default values of a setting
Edit matching criteria	Editing matching criteria
Add or remove matching criteria	Adding or removing matching criteria

Table 12-2. Tasks in polling and thresholds console (Continued)

Tasks	Reference
Edit value of matching criteria	Changing the value of a matching criterion
Modify polling and Polling groups	Modifying polling and polling groups
Control the removal of instrumentation objects	VMware Smart Assurance IP Manager Concepts Guide contains information on optimization of polling settings.

Creating new groups

You can create additional groups within any group category.

Creating a new group enables you to customize the settings for a group of managed objects. Use two methods to create a new group:

- Copy an existing group. The new group contains the same settings and priorities as the original group. Matching criteria and description are not copied. The new group will be added at the last in the priority list of all the groups.
- Create an empty group. The new group does not contain any settings or members. You must add settings and matching criteria, and set the priority of the new group.

  After creating a new group, use the procedures previously described in this chapter to add settings and matching criteria and to set the priority of the new group.

Copying an existing group

To copy an existing group:

1. In the left panel of the Polling and Thresholds Console, right-click the group that you want to copy.
2. Select **Copy** from the pop-up menu to display the **Copy Group** dialog box.
3. In the dialog box, type a name and an optional description for the new group and click **OK**.

   The new group contains the same settings as the group you copied.
4. Edit the settings, matching criteria, and priority of the new group. Change the value of any parameters as necessary.
5. Click **Reconfigure** to implement your changes.

Creating an empty group

To create an empty group:

1. In the left panel of the Polling and Thresholds Console, right-click the top group (for example, Polling Groups).
2. Select **New Group** from the pop-up menu to display the **New Group** dialog box.
3. In the dialog box, type a name and an optional description for the new group and click **OK**.

4 Add settings and matching criteria, and set the priority of the new group. Change the values of any thresholds or parameters as necessary.

5 Click **Reconfigure** to implement your changes.

Modifying the properties of a group

When a group is selected in the left panel of the Polling and Thresholds Console, four tabs appear in the right panel of the console, as shown in Modifying the properties of a group:

- **Settings**
- **Priorities**
- **Matching Criteria**
- **Description**

  By modifying the properties in the Settings, Priorities, or Matching Criteria tab, you can change the configuration of the group. When you finish editing the properties of a group, click **Apply** to save the changes.

Adding or removing Settings parameters

A group's settings determine what parameters are applied to the managed objects that are members of the group.

The **Settings** tab is divided into two sections:

- Current Settings — lists the settings that are applied to the group
- Available Settings — lists additional available settings

  To add or remove a setting:

  a In the left pane of the **Polling and Threshold** console, select the group for which you want to add or remove settings.

  b In the **Settings** tab, select a setting in the **Available Settings** list or in the **Current Settings** list.

  c Click **Add** to move an available setting to the **Current Settings** list, or click **Remove** to move a current setting to the **Available Settings** list.

  d Click **Apply** to save your changes.

  e Click **Reconfigure** to implement your changes.

Modifying the values of Settings parameter

The parameters of a Setting are changed in one of the following two ways:

- By choosing a value from a drop-down menu
- By entering a value in a **Value** field or adjusting a slider bar representing a range of values

Changing the parameters of a setting

To change the parameters of a setting:

1 In the left panel of the **Polling and Thresholds** Console, expand the group and **Settings** for which you want to edit the parameters.

2 Select the specific **Setting** to display the parameters of the selected settings in the right panel of the console.

3 Change the value of a parameter by using one of the following methods:

   For a drop-down menu, click the menu and select a value.

   For a slider bar presentation,

   - Type a value into the **Value** field and press **Enter**.
     - Or, select the slider bar and drag its handle with the mouse to change the value or select the slider bar and use the arrow keys to move its handle to change the value.

4 Click **Apply** to save your changes.

5 Click **Reconfigure** to implement your changes.

Restoring the default values of a setting

To restore the default values of a setting:

1 In the left panel of the **Polling and Thresholds** console, select the particular setting parameter under the **Settings** tab of particular group to display the parameter values of the setting in the right panel.

2 Click **Restore Defaults**.

3 Click **Reconfigure** to implement your changes.

Modifying the priority of groups

In any group category, you can arrange the groups in a high-to-low priority. A candidate member object is compared against the highest priority group first and becomes a member of the first group for which there is a criteria match.

The **Priorities** tab shows groups in the order of their priority, from highest to lowest.

To change the priority of a group:

1 In the **Priorities** tab, select the group for which you want to change the priority.

2 Click the **up** or **down arrow** to change its position relative to the other groups.

3 Click **Apply** to save your changes.

4 Click **Reconfigure** to implement your changes.

Note When a new group is added, by default, the group is added in at the last in the priority list.

Editing matching criteria

A group's matching criteria, which appear at the top of the Matching Criteria tab, are defined by using the attributes of the target class that is associated with the group. Each matching criterion has three fields: Name, Description, and Value.

- Name identifies the name of the target class attribute.
- Description is the description of the target class attribute.
- Value is the value of the target class attribute, which is any combination of text, integers, and wildcards. The Value field for a matching criterion is not case-sensitive.

  As an example of a matching criterion for target class UnitaryComputerSystem, if you specify "Router" for the Type attribute, only routers become members of the group; if you specify "R1*" for the Name attribute, only routers having names beginning with the string R1 become members of the group. Chapter 9, Wildcard Patterns, in the VMware Smart Assurance IP Manager Reference Guide contains information on wildcard patterns.

  When a managed object matches a group's target class and matches all of the attributes defined for the target class, the managed object becomes a member of the group.

Adding or removing matching criteria

To add or remove matching criteria:

1. In the **Disabled Criteria** area of the **Matching Criteria** tab, select a matching criterion.
2. Click **Enable** to make the criterion active, moving it to the top of the **Matching Criteria** tab.

   Use **Disable** to deactivate the criterion, moving it to the **Disabled Criteria** area of the **Matching Criteria** tab.
3. If you are adding a matching criterion, type a matching pattern in the **Value** field.
4. Click **Apply** to save your changes.
5. Click **Reconfigure** to implement your changes.

Changing the value of a matching criterion

To change the value of a matching criterion:

1. In the **Matching Criteria** tab, select the string in the **Value** field or double-click the **Value** field to highlight the current value.
2. Type the new value.
3. Click **Apply** to save your changes.

4 Click **Reconfigure** to implement your changes.

The IP Manager compares a managed object against the matching criteria of the group with the highest priority. If an object matches all the criteria, the IP Manager adds the object as a member of the group. Otherwise, the IP Manager compares the object against the matching criteria of the group with the second highest priority, and so on.

Modifying polling and polling groups

The VMware Smart Assurance IP Manager Deployment Guide provides extensive guidelines on when to make changes to existing polling groups and when to create new polling groups. Consult the deployment guide before modifying any polling groups.

Procedures for modifying existing groups and creating new ones are given in Working with groups and settings.

Performing tasks in the Polling tab

In the Polling tab, you can control the ICMP and SNMP connectivity and performance polling of the managed network objects in your network. IP Manager provides the following connectivity polling and performance polling settings as listed in Connectivity polling and Performance polling settings.

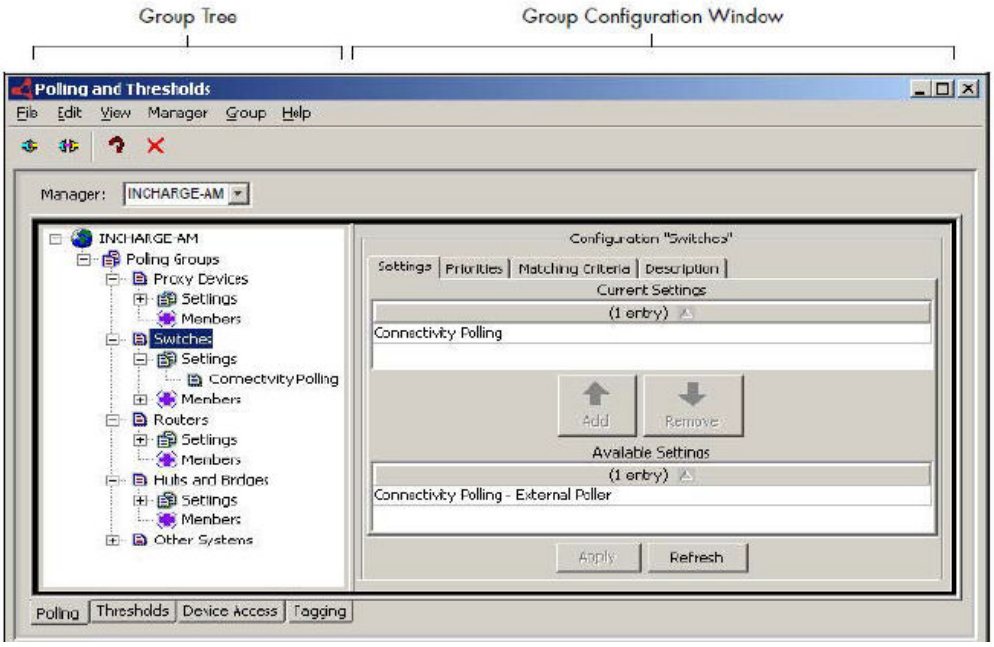
You can enable the ICMP latency monitoring feature as described in How to enable the ICMP latency monitoring feature.

Table 12-3. Connectivity polling and Performance polling settings

Connectivity polling settings	Polling settings for IP performance monitoring
Connectivity polling	Environment Polling
Connectivity polling for IPv6 addresses	Environment Polling - External Poller
Note The Connectivity Polling for IPv6 Addresses setting works in conjunction with the Connectivity Polling setting. Apply the Connectivity Polling for IPv6 Addresses setting to any polling group that has at least one member system with at least one IPv6 address.	FileSystem Polling
	Performance Polling - Ports and Interfaces
	Performance Polling - Processor and Memory
Connectivity Polling-External Poller	
Connectivity Polling for IPv6 addresses-External poller	
Default_IP_Route_Connectivity Polling	
Cleanup Unused Monitoring Instances	

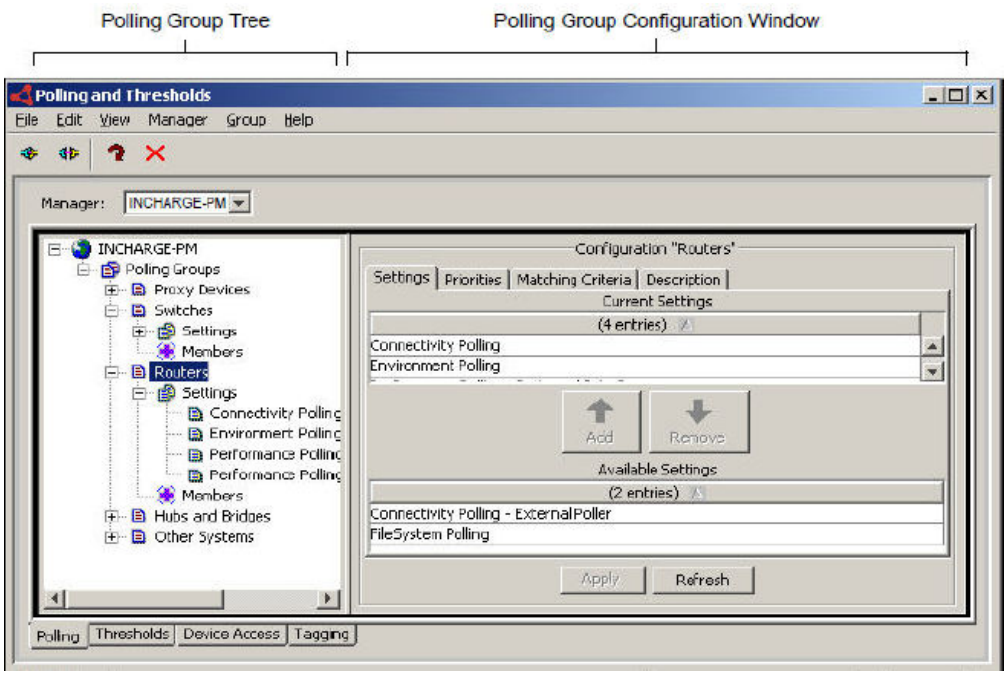
Polling tab for IP Availability Manager shows the Polling tab of a Polling and Thresholds Console connected to an IP Availability Manager instance named INCHARGE-AM. The right panel of the console identifies “Connectivity Polling” as the default setting for the default polling group named “Switches”.

Figure 12-3. Polling tab for IP Availability Manager



Polling tab for IP Performance Manager shows the Polling tab of a Polling and Thresholds Console connected to an IP Performance Manager instance named INCHARGE-PM. The right panel of the console identifies “Connectivity Polling” and “Environment Polling” as two of the four default settings for the default polling group named “Routers”. The other two default settings for Routers are “Performance Polling - Ports and Interfaces” and “Performance Polling - Processor and Memory”.

Figure 12-4. Polling tab for IP Performance Manager



Chapter 7, Polling and Thresholds settings, in the *IP Management Suite Reference Guide* provides more information on Polling settings in IP Availability Manager and IP Performance Manager.

How to enable the ICMP latency monitoring feature

Through the Connectivity Polling setting, you can also enable and configure the ICMP latency monitoring feature by setting the MaximumLatency parameter to a non-zero value. This feature provides round-trip times for all polled IP objects that are monitored by ICMP. The round-trip time reflects the latency, in milliseconds, that is observed during the last ICMP poll.

Prerequisites

The ICMP latency monitoring feature is not enabled by default. It requires the following settings:

1 Set the MaximumLatency parameter to a non-zero number in the Polling and Thresholds Console. Maximum Latency attribute for IP is added under **Settings** > **Connectivity Polling**.

2 Subscribe to the IP::SlowResponseTime event in SAM or in an external subscriber.

Note You need to directly subscribe to the event. Prior to subscribing to the event, ensure that you set the MaximumLatency parameter.

If the round-trip time between the sending of an ICMP request to an IP and the receiving of the ICMP response from the IP exceeds a user-defined latency value, but does not exceed a user-defined timeout value, the IP Availability Manager generates an IP SlowResponseTime event. The Global Manager subscribes to the IP SlowResponseTime event.

Performing tasks in the Thresholds tab

In the **Thresholds** tab, you can set and edit threshold settings as well as policy settings. You can perform the following tasks:

- Control the ICMP and SNMP connectivity and performance polling thresholds and of the managed network objects in your network
- Modify thresholds values and groups
  - Defining thresholds for individual filesystem instances
- Disable Fault Analysis for Interface/Port threshold setting
- *"Policy Settings to control the managed state of interfaces" on page 118*:
  - Applying the Interface Management Policy setting
  - Verifying interface management policy configuration
- VLAN tagging policy:
  - VLAN tagging scenario
  - VLAN tagging naming scheme
  - VLAN Tagging Policy setting
  - Creating a VLAN tag group

  - Verifying VLAN tag group configurations
- Problem Impact Propagation policy over VLANs:
  - VLAN impact propagation scenario
  - VLAN impact propagation group matching criteria
  - Creating VLAN impact propagation group objects
  - Example of creating a VLAN impact propagation group
  - Verifying VLAN impact propagation group configurations

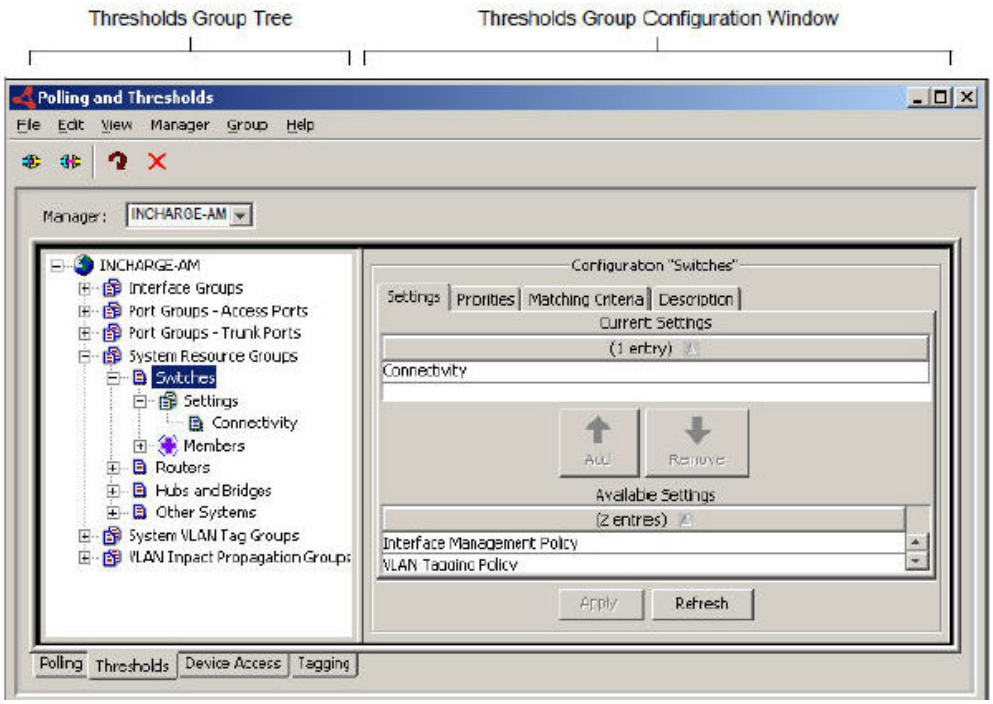
  The IP Manager provides the following threshold settings for IP availability monitoring and IP performance monitoring:

Table 12-4. Thresholds settings for IP availability and performance monitoring.

Thresholds settings for IP availability monitoring	Thresholds settings for IP performance monitoring	Threshold settings for Control
Backup Interface Support	Ethernet Interface/Port Performance	Disable Fault Analysis for Interface/Port threshold setting
Connectivity	Generic Interface/Port Performance	Disable Performance Analysis for Interface/Port threshold setting
Dial-on-Demand Interface Support	Cisco Fast and NvRam Memory	
Interface/Port Flapping	Environment	
	FileSystem	
	Juniper Environment	
	Processor and Memory	
	Temperature settings	
	Voltage settings	

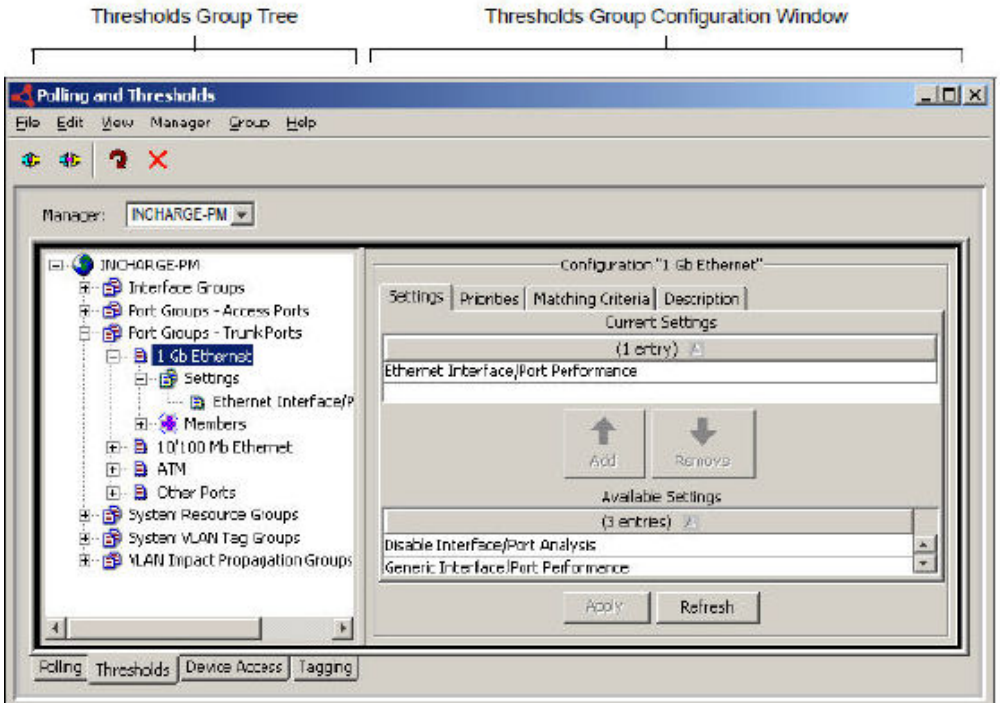
Thresholds tab for IP Availability Manager shows the Thresholds tab in the Polling and Thresholds Console connected to an IP Availability Manager instance named INCHARGE-AM. The right panel of the console identifies "Connectivity" as the default setting for the default "System Resource Group" named "Switches."

Figure 12-5. Thresholds tab for IP Availability Manager



Thresholds tab for IP Performance Manager shows the Thresholds tab of a Polling and Thresholds Console connected to an IP Performance Manager instance named INCHARGE-PM. The right panel of the console Identifies "Ethernet Interface/Port Performance" as the default setting for the default "Port Groups - Trunk Ports" group named "1 Gb Ethernet."

Figure 12-6. Thresholds tab for IP Performance Manager



Instructions for opening the Polling and Thresholds Console are presented in Opening the Polling and Thresholds Console.

Modifying threshold values and threshold groups

The VMware Smart Assurance IP Manager Deployment Guide provides extensive guidelines on when to make changes to existing threshold groups and when to create new threshold groups. Consult the deployment guide before modifying any threshold groups.

Procedures for modifying existing groups and creating new ones are given in Working with groups and settings.

Disable Fault Analysis for Interface/Port threshold setting

The Disable Fault Analysis for Interface/Port setting controls the fault analysis of interfaces and ports. Although available to interface and port threshold groups, this setting is not applied by default to any interface or port group.

To use the Disable Fault Analysis for Interface/Port setting to disable fault analysis for an interface or port group,

1 Consult Adding or removing Settings parameters to add the setting to the Current Settings list.

2 Click **Apply** to save the configuration change.

3 Click **Reconfigure** to implement the configuration change.

Threshold settings for control

Through the following settings, you control the fault and performance analysis of interfaces and ports:

- Disable Fault Analysis for Interface/Port
- Disable Performance Analysis for Interface/Port

Note Interfaces and ports are also known as network adapters.

Disable Fault Analysis for Interface/Port threshold setting

To use the Disable Fault Analysis for Interface/Port setting to disable fault analysis for an interface or port group, consult Adding or removing Settings parameters to add the setting to the Current Settings list, click **Apply** to save the configuration change, and then click **Reconfigure** to implement the configuration change.

Disable Performance Analysis for Interface/Port threshold setting

The Disable Performance Analysis for Interface/Port setting controls the performance analysis of interfaces and ports. Although available to interface and port threshold groups, this setting is not applied by default to any interface or port group.

To use the Disable Performance Analysis for Interface/Port setting to disable performance analysis for an interface or port group, consult Adding or removing Settings parameters to add the setting to the Current Settings list, click **Apply** to save the configuration change, and then click **Reconfigure** to implement the configuration change.

Defining thresholds for individual filesystem instances

You can configure thresholds for individual filesystem instances. A default group called *FileSystem* is available. The utilization thresholds can be configured for each filesystem instance. As a result, a host with multiple filesystems can have different thresholds defined individually for each filesystem instance. Each filesystem instance is monitored individually.

The thresholds defined in the *System Resource Group* serve as the default and may be overwritten by values in one of the new FileSystem groups. If a filesystem instance belongs to a particular group, it inherits the group's threshold settings. In case it does not belong to any group, the thresholds in the System Resource group that the device belongs to will apply.

When new filesystem groups are created, they do not have any matching criteria and are empty. You can add new matching criteria and configure the thresholds for each filesystem group.

1 For the selected filesystem group, change the priority In the Polling and Threshold console > Matching Criteria.

   This can be done in the Matching Criteria tab for the selected filesystem group in the Polling and Thresholds window. Each filesystem group is placed in the order of priority in the Polling and Thresholds window. You can change the priority from the Priority tab. The matching of the filesystem instances begin from the highest priority group and continues in the order of priority. A filesystem instance that matches a group is not considered for matching with the remaining groups.

   Note When creating FileSystem groups, you can use the sample filesystem groups and matching criteria. VMware Smart Assurance IP Manager Reference Guide provides a list of sample filesystem groups and matching criteria.

   Defining thresholds for individual filesystem instances lists the events and the utilization thresholds along with their default values for filesystem.

Table 12-5. Events and thresholds for FileSystem

Events	Utilization threshold	Description of threshold	Default value	Condition
HighUtilization **Note** Severity level: Major	MaxUtilizationPct	The percentage of total size currently in use.	95% **Note** User can change this value for each filesystem instance.	This event is generated when the utilization threshold exceeds the configured threshold value. **Note** This threshold has greater priority over the ModerateUtilization event.
ModerateUtilization **Note** Severity level: Minor	ModerateUtilizationPct	Moderate threshold for percentage of total size currently in use.	80% **Note** User can change this value for each filesystem instance.	This event is generated when the utilization threshold exceeds the configured threshold value.
LowAvailableSpace **Note** Severity level: Minor	MinAvailableSpace	The available space is less than the minimum available space.	1024 **Note** User can change this value for each filesystem instance.	This event is generated when the utilization threshold exceeds the configured threshold value.

External Polling Support for FibreChannelSwitchPort

To enable external polling, do the following in the Polling and Thresholds tab:

1 Remove the Connectivity Polling class from the respective Polling Group where the FibreChannelSwitchPort is present.

2 Add the Connectivity Polling-External Polling setting class for the same Polling Group where the FibreChannelSwitchPort present.

3 Enable InstrumentFibreChannelPort under the Connectivity Polling-External Polling setting classes of the Polling group(s).

4 Apply the reconfiguration to create the default instrumentation class for the FibreChannelSwitchPort.

Policy Settings to control the managed state of interfaces

This section discusses the following system-level policies and the tasks associated with them:

- *"Interface Management Policy" on page 118*
- VLAN tagging policy
- Problem Impact Propagation policy over VLANs

Interface Management Policy

The Interface Management Policy is a system-level setting for controlling the managed state of interfaces. This setting is available to all System Resource Groups and all System VLAN Tag Groups. By default, it is not contained in any of these groups.

If you do not use this setting, all discovered interfaces are automatically managed. Interfaces exist on hosts, routers, and Layer 3 switches.

If you do not want to manage the interfaces that are part of the systems that are members of a system resource threshold group or a system VLAN tag group, apply the Interface Management Policy setting to the group and set the DefaultManagementStatus parameter to UNMANAGED.

Interface Management Policy setting

Default values for the Interface Management Policy setting lists the Interface Management Policy setting parameters.

Table 12-6. Default values for the Interface Management Policy setting

Parameter	Default value	Description
DefaultManagementStatus	MANAGED	Indicates the default management status of interfaces on systems to which this setting applies. The value can be MANAGED or UNMANAGED. Discovery is required for any change to take effect.

Note If you set an interface management policy, the policy can be overridden for a particular interface by using the manage/unmanage options that are available through the Domain Manager Administration Console. Managing the unmanaged physical interface, does not manage its logical interfaces automatically, like IPSec interfaces. Managing the unmanaged group, unmanages its underlying components and this works for Redundancy Groups such as, HSRPGroup, VRRPGroup.

To use the Interface Management Policy setting, add the setting to a system resource threshold group or a system VLAN tag group. Accessing the Polling and Thresholds Console and adding a setting to a group are described in Opening the Polling and Thresholds Console and Working with groups and settings.

When a system becomes a member of a system resource threshold group or a system VLAN tag group to which the Interface Management Policy setting is applied, the IP Manager sets the IsManaged attribute of the interfaces of the system in accordance to the DefaultManagementStatus parameter value:

- DefaultManagementStatus = UNMANAGED sets IsManaged to FALSE.
- DefaultManagementStatus = MANAGED sets IsManaged to TRUE.

Applying the Interface Management Policy setting

The following procedure demonstrates how to use the Polling and Thresholds Console to apply the Interface Management Policy setting to the default system resource group named Routers:

1. In a **Polling and Thresholds Console** attached to an IP Availability Manager or IP Performance Manager, click the **Thresholds** tab to display the thresholds group tree.
2. In the thresholds group tree, click the plus signs (**+**) to expand the thresholds group tree as shown in the following display, then click **Routers** to display the configuration window and the **Settings** tab.
3. In the **Available Settings** list of the **Settings** tab, select **Interface Management Policy** and then click **Add** to move the **Interface Management Policy** setting to the **Current Settings** list.
4. Click **Apply** to save your changes.
5. In the thresholds group tree, click **Interface Management Policy** to display its setting configuration window.
6. In the setting configuration window, click the drop-down menu and select **MANAGED** or **UNMANAGED** from the menu; MANAGED is the default.
7. Click **Apply** to save your changes.
8. Optional: In the thresholds group tree, click **Routers** to display the configuration window.

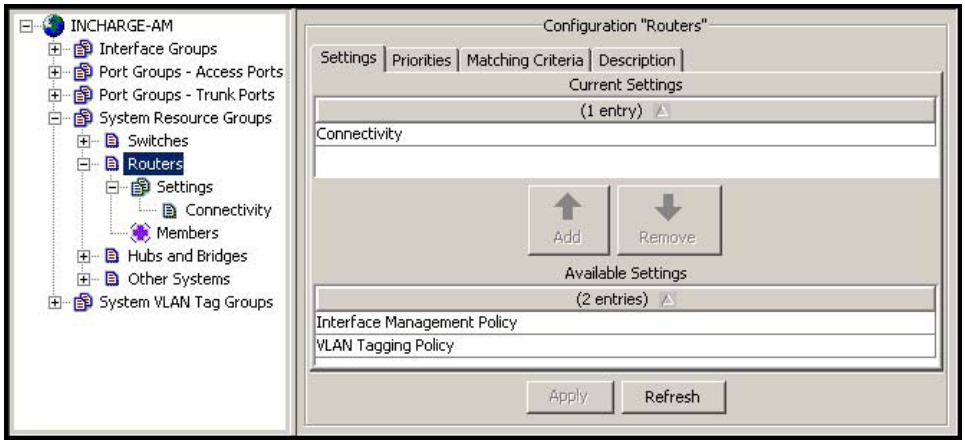
9 Optional: In the configuration window, click **Matching Criteria** to display the **Matching Criteria** tab.

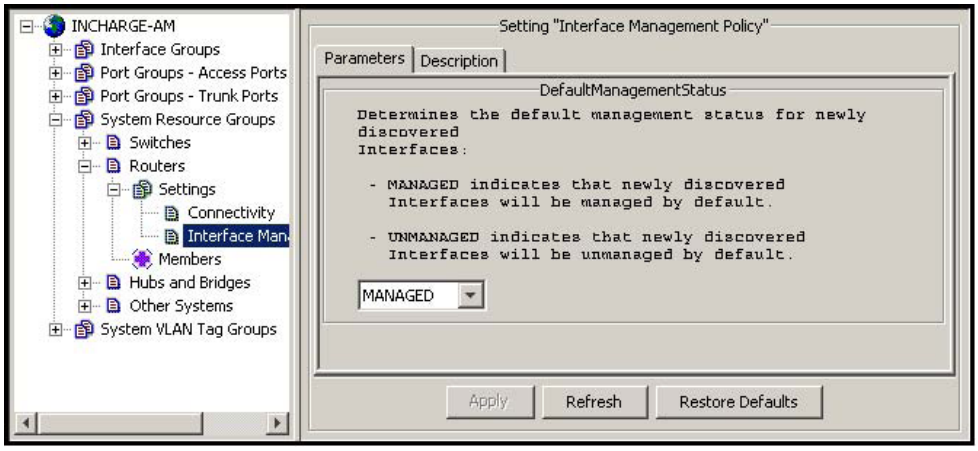
10 Optional: In the **Matching Criteria** tab, specify additional matching criteria for the Routers group and click **Apply** to save your changes. The procedure for editing matching criteria is given in Editing matching criteria.

The IsManaged attribute value of the interfaces of any router that becomes a member of the Routers group will automatically be set to the value that you selected in step 6:

- DefaultManagementStatus = UNMANAGED will set IsManaged to FALSE.
  - DefaultManagementStatus = MANAGED will set IsManaged to TRUE.

11 Click **Reconfigure** to implement your changes.





Verifying interface management policy configuration

To verify that the interface management policy of your Routers group is configured properly:

1 Attach the Domain Manager Administration Console to the target IP Manager.

Instructions for opening the Domain Manager Administration Console are presented in Opening the Global Console.

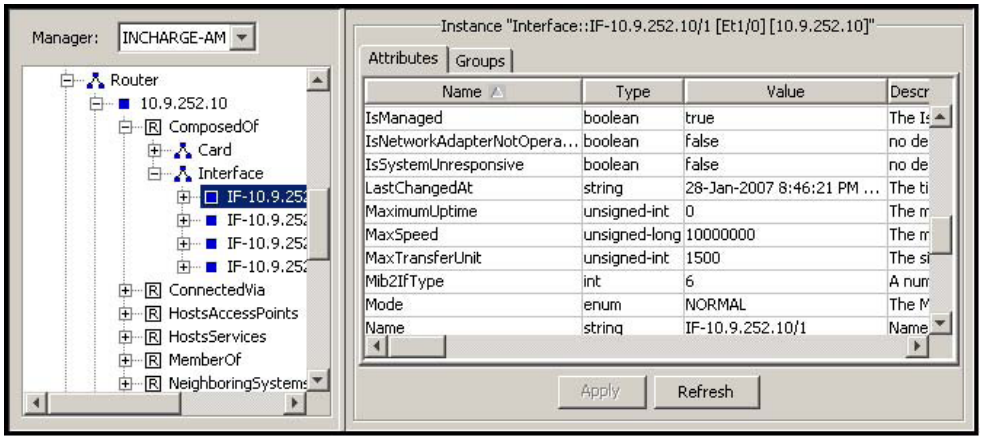
2 In the console, select **Topology** > **Discover All** to rediscover all managed systems.

3 In the left panel of the **Domain Manager Administrator Console**, expand the **Router** class to see the router instances that the IP Manager just rediscovered.

4 Select a router instance that is configured to be a member of the Routers group, expand its **ComposedOf** relationship, expand the **Interface** class, and select an interface instance to display its **Attributes** tab in the right panel.

5 In the **Attributes** tab, verify that the value of the IsManaged attribute is set to the expected value.

6 Repeat steps 4 and 5 for each interface instance of each router instance that is configured to be a member of the Routers group.



VLAN tagging policy

The VLAN Tagging Policy is a system-level setting for ensuring that systems are assigned to their appropriate VLAN. By default, this setting is contained in the System VLAN Tag Group named Switches, is available to the System VLAN Tag Group named Other Systems, and is available to all System Resource Groups.

A networked environment may consist of many separate networks, each having many VLANs. One VLAN may have the same name (VLAN ID) as a VLAN in another network. Identically named VLANs do not create problems across different networks; however, when these VLANs are discovered and placed into a single topology, conflicts arise.

If you do not use the VLAN Tagging Policy setting, the IP Manager will combine any discovered VLANs that have identical names into a single VLAN of that name; for example, VLAN-1. And systems that belong to the identically named but locally distinct VLANs will be represented in the modeled topology as belonging to VLAN-1. Because of this incorrect representation in the modeled topology, the IP Manager root-cause and impact analysis will produce inaccurate results.

If you do use the VLAN Tagging Policy setting, the IP Manager will distinguish any discovered distinct VLANs that have identical names by adding a unique tag to the name of each of the distinct VLANs. And systems that belong to identically named but distinct VLANs will be represented in the topology as belonging to distinct VLANs. Because of this correct representation in the modeled topology, the IP Manager root-cause and impact analysis will produce accurate results.

VLAN tagging scenario

Consider the following scenario:

- In the managed network environment, two switches (SW1, SW2) in one VLAN and two switches (SW2, SW3) in another VLAN are members of two locally distinct VLANs that have the same name, VLAN-1.
- Without creating a VLAN tag group for at least one of the VLANs, the IP Manager will combine the VLANs into one VLAN, named VLAN-1, that includes all four switches.
- By creating a VLAN tag group for at least one of the VLANs, the IP Manager will assign SW1 and SW2 to one VLAN, and assign SW2 and SW3 to another VLAN.
- For example, if you create a VLAN tag group that has a tag named “Marketing” and having matching criteria that allow SW1 and SW2 to become members of that group, the IP Manager will assign SW1 and SW2 to a VLAN named VLAN-Marketing-1, and assign SW2 and SW3 to a VLAN named VLAN-1.

VLAN tagging naming scheme

Develop a naming scheme for the tags and decide which systems to associated with which tags. Tagged VLANs appear in the topology as:

```
VLAN-<tag>-<VLAN number>
```

VLAN Tagging Policy setting

Default values for the VLAN Tagging Policy setting lists the VLAN Tagging Policy setting parameters.

Table 12-7. Default values for the VLAN Tagging Policy setting

Parameter	Default value	Description
Tag	Original configuration (no tag)	Determines the tag for the VLANs configured on the systems that belong to a particular system VLAN tag group. If the tag is specified, the VLANs will be named as VLAN-*<tag>*-*<VLAN number>*. For example, if you specify a tag, “Marketing,” for an originally configured VLAN-1, it appears as VLAN-Marketing-1. Discovery is required for any change to take effect.

To use the VLAN Tagging Policy setting to create a system VLAN tag group:

- You need to assign a tag to the setting’s Tag parameter.
- Accessing the Polling and Thresholds Console and modifying the parameters in a setting are described in Opening the Polling and Thresholds Console andWorking with groups and settings.

Creating a VLAN tag group

The following procedure demonstrates how to use the Polling and Thresholds Console to create a VLAN tag group for switches by assigning a tag and adding additional matching criteria to the default System VLAN Tag Group named Switches:

1 In a **Polling and Thresholds** console that is attached to the IP Availability Manager or the IP Performance Manager, click the **Thresholds** tab to display the thresholds group tree.

2 In the thresholds group tree, click the plus signs (**+**) to expand the thresholds group tree as shown in the following display, then click **VLAN Tagging Policy** to display its setting configuration window.

3 In the setting configuration window, type a tag value that is meaningful for your configuration and press **Enter**.

Note For **Apply** to be available, you must press the **Enter** key (on your keyboard) after typing or changing the tag value.

4 Click **Apply** to save your changes.

5 In the thresholds group tree, click **Switches** to display the configuration window.

6 In the configuration window, click **Matching Criteria** to display the **Matching Criteria** tab.

7 In the **Matching Criteria** tab, specify the additional matching criteria that will allow only the switches that belong to the target VLAN to become members of this group. The procedure for specifying matching criteria is given in Editing matching criteria.

Note Specify matching criteria broad enough so that *all* switches that belong to the target VLAN become members of this group. Otherwise, the IP Manager will create multiple names (multiple VLAN instances) for the target VLAN, which will produce incorrect correlation analysis results.

For example, if a distinguishing feature for the switches that belong to the target VLAN is that their names all contain the string “sw,” set the **Name** attribute to ***sw*** to ensure that only those switches become members of the Switches group.

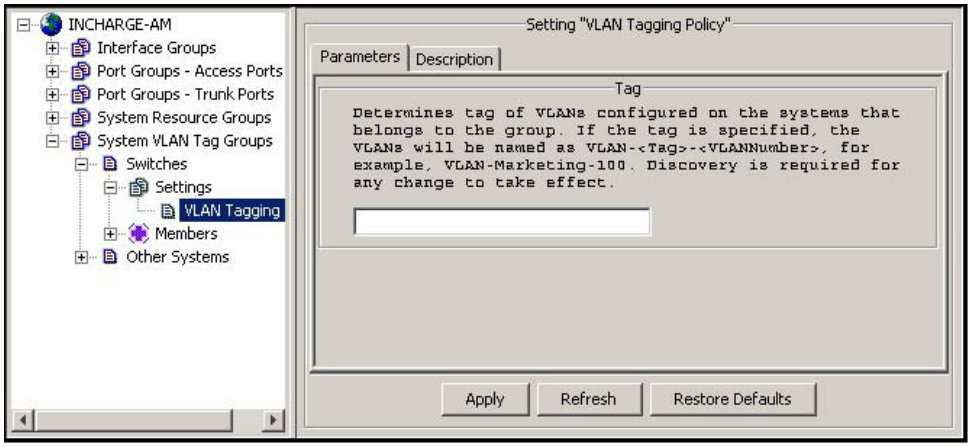
For any switch that becomes a member of this group, the name of any VLAN to which the switch belongs will be tagged with the value that you specified in step 3 .

8 Click **Apply** to save your changes.

9 Click **Reconfigure** to implement your changes.

Create a system VLAN tag filter for each group of systems that belong to identically named but locally distinct VLANs.

To create additional system VLAN tag groups for switches, create new groups from the Switches group configured in the preceding procedure and repeat that procedure for the newly created groups. The procedure for creating new groups is given in Creating new groups. To create system VLAN tag groups for routers, hubs, or bridges, start with the default system VLAN tag group named Other Systems.



Verifying VLAN tag group configurations

To verify that your system VLAN tag groups are configured properly:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   Instructions for opening the Domain Manager Administration Console are presented in Chapter 3, “Opening the Global Console.”

2 In the console, select **Topology** > **Discover All** to rediscover all managed systems.

3 In the topology tree of the console, expand the **VLAN** class to see the tagged VLAN instances (and their member systems) that the IP Manager just created.

4 Verify that the rediscovered switches are members of the appropriate VLANs.

Problem Impact Propagation policy over VLANs

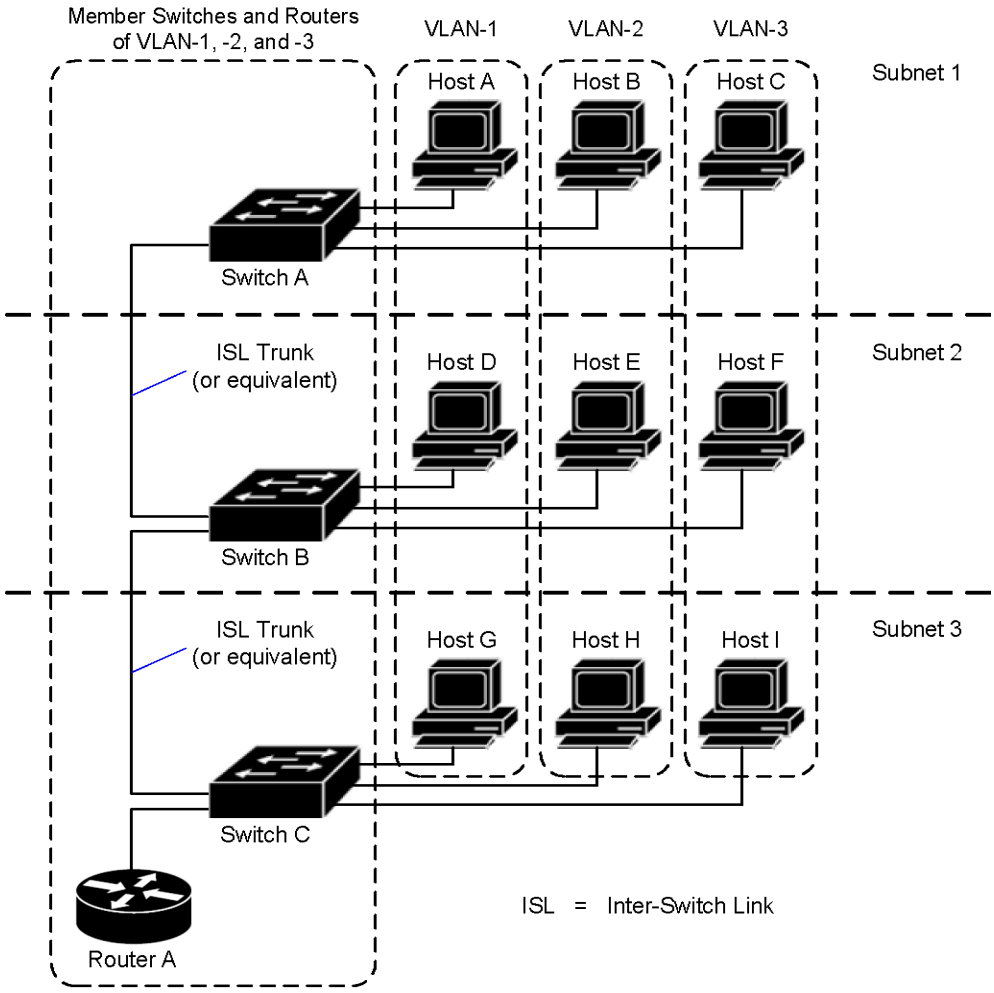
The “Problem Impact Propagation policy over VLANs” is a VLAN-level setting for preventing the generation of impacts of failed devices in a VLAN on other devices in the VLAN. By default, this setting is contained in the VLAN Impact Propagation Group named VLANs and is available to the VLAN Impact Propagation Group named Other VLANs.

For two failed devices that are logically connected through a common VLAN, the IP Manager will not only create problem notifications for the failed devices, but will also create impact notifications that claim that each failed device is responsible for the failure of the other failed device. While, in general, this analysis is correct, customers might want to prevent this type of impact propagation in certain VLANs, such as in management VLANs or in default VLANs (for example, in the default VLAN-1 for Cisco switches and routers). The “Problem Impact propagation policy over VLANs” setting allows customers to accomplish this end.

VLAN impact propagation scenario

VLANs that span multiple switches and multiple subnets shows three VLANs spanning three subnets.

Figure 12-7. VLANs that span multiple switches and multiple subnets



In VLANs that span multiple switches and multiple subnets, Switches A, B, and C provide three separate bridge groups, one for each VLAN, and Router A provides routing for inter-VLAN communications.

If Switch A fails, the IP Manager creates (and the Global Manager imports) the following notifications:

- One Switch A Down (problem) notification
- One Switch A Unresponsive (event) notification
- One TrunkCable DownOrFlapping (event) notification
- Three Cable DownOrFlapping (event) notifications—assuming Hosts A, B, and C are “managed”

  The IP Manager uses the events as symptoms to diagnose the Switch A Down problem. The events will appear in the Impact tab of the Notification Properties dialog box for the Switch A Down problem.

  Note Events that are used as symptoms to diagnose a failure can also be perceived as impacts that result from the failure.

  If Switch B also fails, the IP Manager proceeds as follows:

- Clears its original Switch A Down notification, creates a new Switch A Down notification, and creates a Switch B Unresponsive notification; the Impact tab of the Notification Properties dialog box for the new Switch A Down problem will list an additional impact: Switch B Unresponsive.
- Creates a Switch B Down notification, a Switch B Unresponsive notification, and five TrunkCable/Cable DownOrFlapping notifications; the Impact tab of the Notification Properties dialog box for the Switch B Down problem will list a Switch B Unresponsive impact, five TrunkCable/Cable DownOrFlapping impacts, and a Switch A Unresponsive impact.

  The Switch A and B Down notifications claim that each switch is responsible for the associated failure of the other. In this example, the switches have an impact on each other because they are members of the same set of VLANs: VLAN-1, VLAN-2, and VLAN-3.

  Creating a VLAN impact propagation group that allows VLAN-1, VLAN-2, and VLAN-3 to become members of the group will prevent the linking of the Switch B Unresponsive impact to the Switch A Down problem, and will prevent the linking of the Switch A Unresponsive impact to the Switch B Down problem.

VLAN impact propagation group matching criteria

The matching criteria for a VLAN impact propagation group are the VLAN class and two of its attributes:

- CreationClassName
- VLANKey

  By default, the value of the CreationClassName attribute is set to “VLAN,” and the value of the VLANKey attribute is undefined. The value that you specify for the VLANKey attribute determines which VLANs become members of a VLAN impact propagation group.

The value of the VLANKey attribute is included in the value of the Name attribute for a VLAN object in the following way:

```
VLAN-<VLANKey>
```

The value of the VLANKey attribute distinguishes a VLAN from all other VLANs in the managed topology and consists of any combination of the following identifiers:

```
<management domain name> <user-defined VLAN tag> <VLAN number (ID)>
```

An example of a VLANKey value consisting of just a management domain name is:

```
VLAN #66
```

An example of a VLANKey value consisting of a user-specified VLAN tag and a VLAN number is:

```
Marketing-1
```

An example of a VLANKey value consisting of just a VLAN number is 1.

Creating VLAN impact propagation group objects

Currently, the underlying software does not automatically create the default group objects of the “Problem Impact propagation policy over VLANs” setting, which means that the group category “VLAN Impact Propagation Groups” does not automatically appear on the Thresholds tab of a Polling and Thresholds Console that is attached to a IP Manager. Instead, a user must issue an sm_adapter command to manually create the VLAN impact propagation group objects. Once created, the group objects are available every time that the IP Manager is restarted.

To create the default group objects of the "Problem Impact propagation policy over VLANs" setting:

1 Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command on one line and then press **Enter.**

```
sm_adapter -s <
   IP Manager

 instance name

   instance name> -f <full path to conf/discovery/settings.conf file
>


      icf/init-settings.asl
```

Option	Description
-f <path>	Full path to the settings.conf file
icf/init-settings.asl	Script that parses the settings.conf file

For example:

```
sm_adapter -s INCHARGE-AM -f /opt/InCharge/IP/smarts/conf/discovery/settings.conf icf/init-settings.asl
```

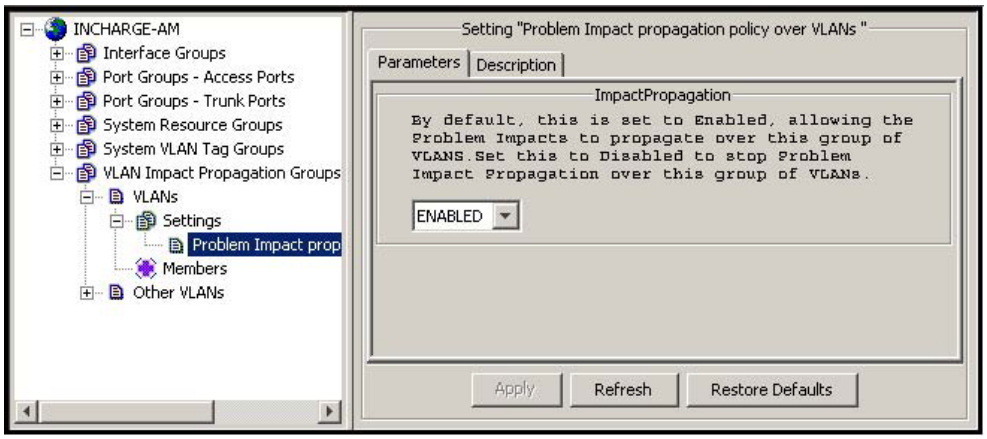
Example of creating a VLAN impact propagation group

The following procedure demonstrates how to use the Polling and Thresholds Console to create a VLAN impact propagation group by disabling impact propagation and adding matching criteria to the default VLAN Impact Propagation Group named VLANs:

1 In a Polling and Thresholds Console that is attached to an IP Availability Manager or IP Performance Manager, click the **Thresholds** tab to display the thresholds group tree.

2 In the thresholds group tree, click the plus signs (**+**) to expand the **VLAN Impact Propagation Groups** as shown in the following display, and then click **Problem Impact propagation policy over VLANs** to display its setting configuration window.

Figure 12-8. Setting "Problem Impact propagation policy over VLANs"



3 In the setting configuration window, click the drop-down menu and select **DISABLED** from the menu; ENABLED is the default.

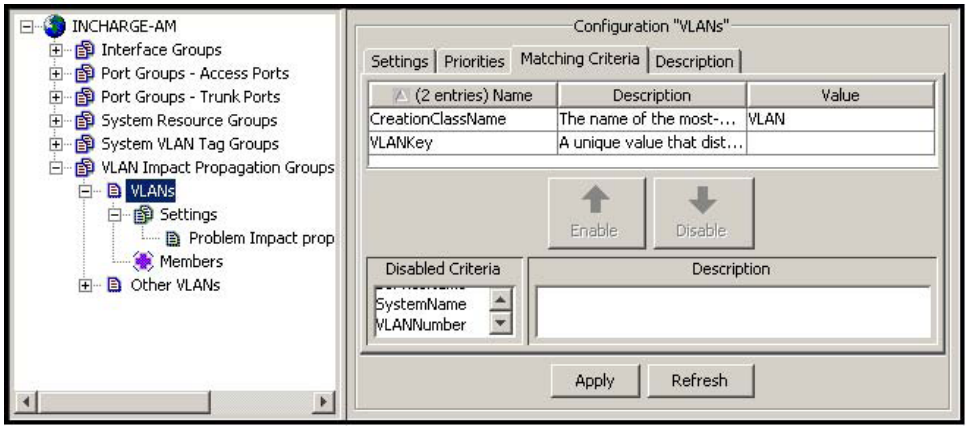
4 Click **Apply** to save your changes.

5 In the thresholds group tree, click **VLANs** to display the configuration window.

6 In the configuration window, click **Matching Criteria** to display the **Matching Criteria** tab.

7 In the **Disabled Criteria** list of the **Matching Criteria** tab, select **VLANKey** and then click **Enable** to move the **VLANKey attribute** to the top of the **Matching Criteria** tab.

Figure 12-9. VLAN Configuration



8 Double-click the **Value** field of the VLANKey attribute and type the matching criteria that will allow only the target VLANs to become members of this group. The procedure for specifying matching criteria is given in Editing matching criteria.

Note Specify matching criteria broad enough so that all VLANs supported by a particular group of switches and routers become members of this group. Otherwise, the IP Manager will continue to propagate impacts over the VLAN or VLANs that do not become a member of this group.

For example, if the VLANs supported by a particular group of switches have VLANKey values containing the string “1,” “2,” or “3”(as in the example given in VLAN Configuration), set the **VLANKey** value to ***1|2|3** to ensure that all of those VLANs become members of the VLANs group.

The isImpactPropOverVLANEnabled attribute value of any VLAN that becomes a member of the VLANs group will automatically be set to the value that you selected in step 3:

- ImpactPropagation = DISABLED will set isImpactPropOverVLANEnabled to FALSE.
  - ImpactPropagation = ENABLED will set isImpactPropOverVLANEnabled to TRUE.

9 Optional: In the **Matching Criteria** tab, specify additional matching criteria for the VLANs group.

10 Click **Apply** to save your changes.

11 Click **Reconfigure** to implement your changes.

Create a VLAN propagation impact group for each group of VLANs for which you want to disable impact propagation.

To create additional VLAN impact propagation groups, create new groups from the VLANs group configured in the preceding procedure and repeat that procedure for the newly created groups. To create VLAN impact propagation groups for VLANs that have a CreationClassName attribute value other than “VLAN,” start with the default VLAN impact propagation group named Other VLANs.

Verifying VLAN impact propagation group configurations

To verify that your VLAN impact propagation groups are configured properly:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   Instructions for opening the Domain Manager Administration Console are presented in Opening the Domain Manager Administration Console.

2 In the console, select **Topology** > **Discover All** to rediscover all managed systems.

3 In the topology tree of the console, expand the **VLAN** class to see the VLAN instances that the IP Manager just rediscovered.

4 Select a VLAN instance that is configured to be a member of a VLAN impact propagation group and click its name to display its Attributes tab in the right panel.

5 In the **Attributes** tab, verify that the value of the isImpactPropOverVLANEnabled attribute is set to FALSE.

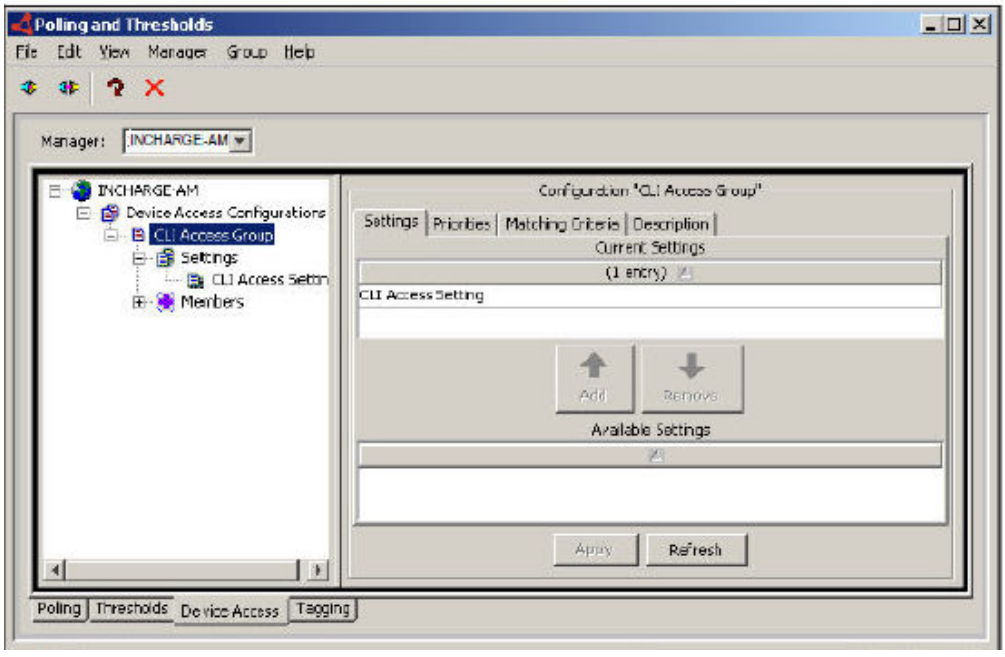
6 Repeat steps 4 and 5 for each VLAN instance that is configured to be a member of a VLAN impact propagation group.

Performing tasks in the Device Access tab

In the Device Access tab, you can:

- Create a device access group as described in Creating a device access group
- Modify tagging filters as described in Modifying tagging filter values and creating tagging filter groups

Figure 12-10. Device Access tab



Device Access tab shows the Device Access tab of a Polling and Thresholds Console connected to an IP Availability Manager instance named INCHARGE-AM. The right panel of the console identifies "CLI Access Setting" as the default setting for the default polling group named "CLI Access Group."

The Device Access tab of the Polling and Thresholds Console connected to an IP Performance Manager, or connected to IP Availability Manager and IP Performance Manager running as a single process, is identical to what is shown in Device Access tab. The device access group and setting configurations are common to all IP Managers started from the same IP Manager installation area.

Modifying device access values and creating device access groups

Currently, the Device Access tab provides only one default group and one default setting, which are used to create CLI login credential sets. Procedures for modifying existing device access groups and creating new ones are given in Creating a device access group and Working with groups and settings.

Default device access groups and settings

Through the default device access groups, you create sets of credentials (such as login credentials) needed to access systems in the managed network environment.

Default device access groups

The Device Access tab of the Polling and Thresholds Console provides one default device access group named CLI Access Group.

Default device access groups lists the matching criteria and the default setting for CLI Access Group. The matching-criteria target class for CLI Access Group is UnitaryComputerSystem.

Table 12-8. Default device access groups

Device access group	Matching criteria	Default settings
CLI Access Group	UnitaryComputerSystem attributes: ■ Name = *	CLI Access Setting

Creating a device access group

In an environment managed by Network Protocol Manager for EIGRP and Network Protocol Manager for IS-IS, you, as the Network Protocol Manager administrator, must create a CLI access group for each group of managed Cisco routers configured with a different access protocol or a different set of CLI login credentials.

The assumption is that all routers that are using a particular access protocol are configured with the same set of credentials (and time-out value). If, in fact, there are two groups of routers that are using a particular access protocol, each configured with its own set of credentials, you would create two CLI access groups for that particular access protocol; for example, create two CLI access groups named Telnet_S1 and Telnet_S2. The matching criteria defined for each of the groups limits the group's members to the appropriate routers.

The basic steps for creating and configuring CLI access groups are:

1 In a Polling and Thresholds Console that is attached to an IP Availability Manager or IP Performance Manager, click the **Device Access** tab to display the device access group tree.

2 In the device access group tree, use the **CLI Access Group** to create a CLI access group for each group of routers that have a different access protocol or a different set of CLI login credentials. One way to create a new CLI access group is to right-click **CLI Access Group** and select **Copy**. (The procedure for creating groups is given in “Creating new groups” on page 147.)

3 Edit the parameters as appropriate in each of the CLI access groups.

4 Define matching criteria for each of the CLI access groups to limit each group’s membership to the router or group of routers for which the parameters that are specified in step 3 apply.

   Start by defining Vendor = <*vendor _name*>. For example,

```
Vendor = Cisco
Name = R1*
```

5 Change the priority of the CLI access group as needed. Typically, the default CLI Access Group is placed at the bottom of the priority list. (The procedure for changing the priority of groups is given in “Modifying the priority of groups” on page 145.)

   Modify the properties of a CLI access group in the same way that you modify the properties of a polling or threshold group: through the Settings, Priorities, and Matching Criteria tabs, available when the group name is selected in the left panel of the Polling and Thresholds Console. After customizing the properties of a group, click Apply to save the changes. Click **Reconfigure** or select **Reconfigure** from the Group menu to make the configuration changes take effect.

   Upon adding the IP Availability Manager as a source, Network Protocol Manager imports the CLI access groups and the EIGRP or IS-IS object set from IP Availability Manager. For each Cisco router in the object set, Network Protocol Manager searches through the prioritized list of CLI access groups for a router match, finds a match, and then uses the associated access protocol and credentials to log in to the Cisco router and gather EIGRP or IS-IS information from the router. If no match is found, Network Protocol Manager writes an error to its log file.

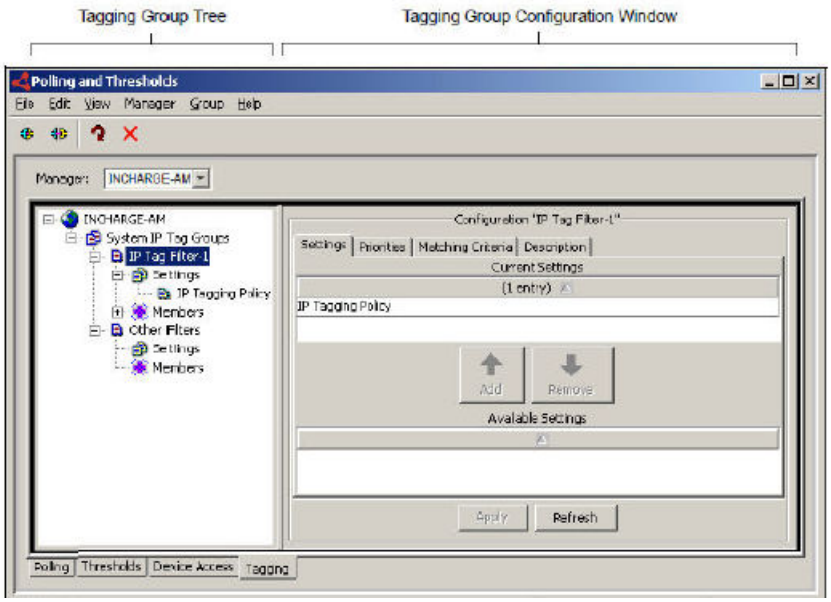
Performing tasks in the Tagging tab

In the IP Tagging tab, you can:

- Create IP tag filter groups as described in Creating an IP tag filter group.

- Verify an IP tag filter group as described in Verifying IP tag group configurations.

Figure 12-11. Tagging tab



Tagging tab shows the Tagging tab of a Polling and Thresholds Console connected to an IP Availability Manager instance named INCHARGE-AM. The right panel of the console identifies "IP Tagging Policy" as the default setting for the default tag group named "IP Tag Filter-1."

The Tagging tab of the Polling and Thresholds Console connected to an IP Performance Manager, or connected to IP Availability Manager and IP Performance Manager running as a single process, is identical to what is shown in Tagging tab. The tag group and setting configurations are common to all IP Managers started from the same IP Manager installation area.

Modifying tagging filter values and creating tagging filter groups

Procedures for modifying existing tagging filter groups and creating new ones are given in:

- Modifying tagging filter values and creating tagging filter groups
- Creating an IP tag filter group

Default tagging filter groups and settings

Using the default tagging filter groups for the IP Manager, you create IP tag filter groups to control the naming of the IP addresses gathered from the systems in the managed network environment.

Default tagging groups

The IP Manager provides two default tagging groups:

- IP Tag Filter-1

- Other Filters

  Tagging groups and matching criteria lists the default tagging groups, their matching criteria, and their settings. The matching-criteria target class for tagging groups is UnitaryComputerSystem.

Table 12-9. Tagging groups and matching criteria

Tagging group	Matching criteria	Default settings
IP Tag Filter-1	UnitaryComputerSystem attributes: ■ Name = *	IP Tagging Policy
Other Filters	UnitaryComputerSystem attributes: ■ No attributes	IP Tagging Policy

The Other Filters tag group does not contain any matching-criteria attributes and has the lowest priority. Systems that do not match the criteria for the IP Tag Filter-1 tag group become members of the Other Filters tag group.

Default tagging settings

A single default tagging setting named IP Tagging Policy is provided by the IP Availability Manager. The IP Tagging Policy setting configures the IP tagging for a system. Default values for the IP Tagging Policy setting lists the IP Tagging Policy setting parameters, and Parameters of default setting IP Tagging Policy shown in a single view shows all of the parameters in a single view.

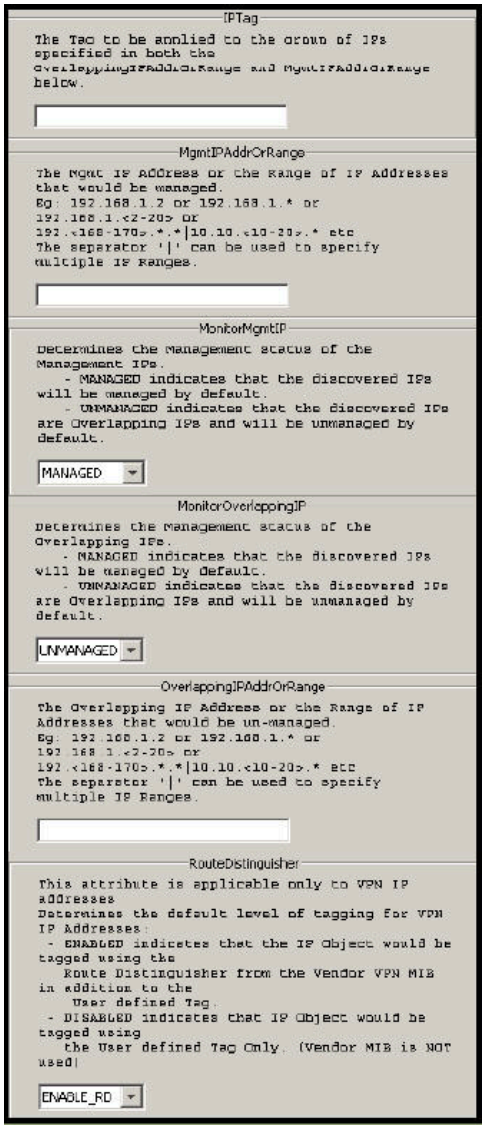
Table 12-10. Default values for the IP Tagging Policy setting

Parameter	Value	Description
IPTag	String of unspecified length Default: Null string (empty)	Specifies the name to be assigned to this tag. This tag will be applied to the IP addresses falling within the ranges that are specified for the OverlappingIPAddrOrRange and MgmtIPAddrOrRange parameters.
MgmtIPAddrOrRange	String of unspecified length Default: Null string (empty)	Specifies the IP address or range of IP addresses to be managed by default. Use the pattern matching criteria in Appendix 15, “Wildcard Patterns.” For example: ■ 192.168.1.2 or 192.168.1.* ■ 192.168.1.<2-20> ■ 192.<168-170>.*.*|10.10.<10-20>.* The pipe (|) separator can be used to specify multiple IP ranges.
MonitorMgmtIP	MANAGED,UNMANAGED Default: MANAGED	Determines the management status to be assigned to the IP addresses falling within the range of the MgmtIPAddrOrRange parameter: ■ MANAGED Indicates that the discovered IP addresses are to be managed. ■ UNMANAGED Indicates that the discovered IP addresses are to be unmanaged.

Table 12-10. Default values for the IP Tagging Policy setting (Continued)

Parameter	Value	Description
MonitorOverlappingIP	MANAGED,UNMANAGED Default: UNMANAGED	Determines the management status to be assigned to the IP addresses falling within the range of the OverlappingIPAddrOrRange parameter: ■ MANAGED Indicates that the discovered IP addresses are to be managed. ■ UNMANAGED Indicates that the discovered IP addresses are to be unmanaged.
OverlappingIPAddrOrRange	String of unspecified length Default: Null string (empty)	Specifies the IP address or range of IP addresses to be unmanaged by default. Use the pattern matching criteria in Appendix 15, “Wildcard Patterns.” For example: ■ 192.168.1.2 or 192.168.1.* ■ 192.168.1.<2-20> ■ 192.<168-170>.*.*|10.10.<10-20>.* The pipe (|) separator can be used to specify multiple IP ranges.
RouteDistinguisher	ENABLE_RD,DISABLE_RD Default: ENABLE_RD	Determines the level of tagging for VPN-IP (VRF IP) addresses: ■ ENABLE_RD Indicates that the IP objects created for the VRF IP addresses falling within the range that is specified for the OverlappingIPAddrOrRange parameter are to be tagged by using the route distinguisher from the vendor VPN MIB, in addition to the user-defined tag that is specified in the IPTag parameter. ■ DISABLE_RDIndicates that the IP objects created for the VRF IP addresses falling within the range that is specified for the OverlappingIPAddrOrRange parameter are *not* to be tagged by using the route distinguisher from the vendor VPN MIB; only the user-defined tag that is specified in the IPTag parameter is to be used to tag these IP objects.

Figure 12-12. Parameters of default setting IP Tagging Policy shown in a single view



Creating an IP tag filter group

An IP tag filter group has two sets of matching criteria:

- System

  The membership class for an IP tag filter group is UnitaryComputerSystem. Only the discovered systems matching the matching criteria (attributes that are specified for UnitaryComputerSystem) for a particular group can become members of that group.

- IP address range

  The IP address range matching criteria consists of two IP address ranges: one for management IP addresses and one for overlapping IP addresses. Each of the ranges may contain a single IP address. Only the IP addresses collected for the discovered system matching either of the IP address ranges will be tagged in accordance to the user-specified tag for the group.

If a system (device) matches the system criteria for the filter, all IP addresses collected for the discovered system are subject to the management IP address range and the overlapping IP address range defined for the filter. If a system does not match the system criteria for the filter, none of the IP addresses collected for that discovered system are subject to the two IP address ranges defined for the filter.

If an IP address falls within the management IP address range, the user-specified tag for the filter is applied to the IP object created for that IP address. If the IP address does not fall within that range, the user-specified tag for the filter is not applied to the IP object created for that IP address:

- Any management IP object created for an IP address matching the management IP address range is MANAGED by default. You can change the value to UNMANAGED by selecting UNMANAGED from the MonitorMgmtIP drop-down menu.
- Any management IP address created for an IP address matching the management IP address range is excluded from the overlapping IP address range defined for the IP tag filter group.

  If an IP address falls within the overlapping IP address range, the user-specified tag for the filter is applied to the IP object created for that IP address. If the IP address does not fall within that range, the user-specified tag for the filter is not applied to the IP object created for that IP address:

- Any overlapping IP object created for an IP address matching the overlapping IP address range is UNMANAGED by default. You can change that value to MANAGED by selecting MANAGED from the MoinitorOverlappingIP drop-down menu.

  The RouteDistinguisher parameter (ENABLE_RD or DISABLE_RD; ENABLE_RD by default) for an IP tag filter group is applicable to all VRF IP addresses that fall within the overlapping IP address range defined for the filter. For an enabled RouteDistinguisher parameter, any IP address falling within the range of the overlapping IP address range is tagged with the route distinguisher name that is associated with the VRF to which that IP address belongs. (The IP Manager gathers route distinguisher information by reading the vendor VPN MIB on the discovered system.) Conversely, if the RouteDistinguisher parameter is disabled, any VRF IP address falling within the overlapping IP address range is *not* tagged with the route distinguisher value.

Note VRF IP addresses *not* falling within the overlapping IP address range will be tagged with the route distinguisher value regardless of the RouteDistinguisher parameter setting (ENABLE_RD, DISABLE_RD). The RouteDistinguisher parameter setting applies only to the VRF IP addresses that fall within the overlapping IP address range defined for the filter.

Single tagging and double tagging of an IP object created for a discovered IP address are both possible. “IP tag naming scheme” on page 216 provides descriptions and examples of single- and double-tagged IP objects.

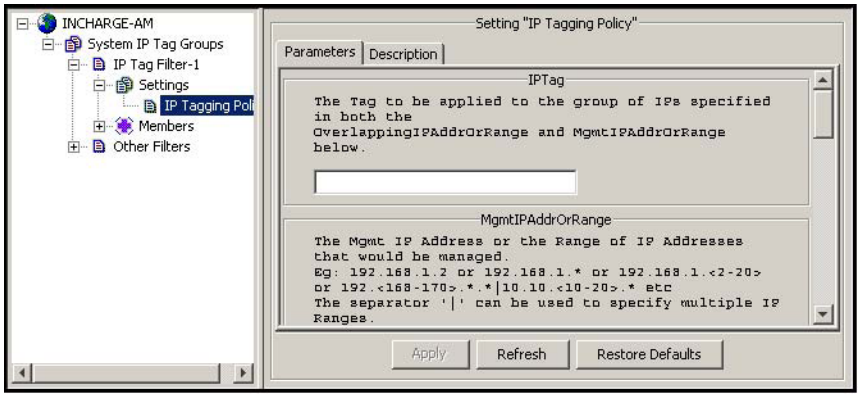
Example of creating an IP tag filter group

The following procedure demonstrates how to use the Polling and Thresholds Console to create an IP tag filter group for routers by assigning values (by using the IP Tagging Policy setting) and adding additional matching criteria to the default system IP tag group named IP Tag Filter-1:

1 In a Polling and Thresholds Console that is attached to an IP Availability Manager or IP Performance Manager, click the **Tagging** tab to display the tagging group tree.

2 In the tagging group tree, click the plus signs (**+**) to expand the tagging group tree as shown in Example of IP tagging policy, then click **IP Tagging Policy** to display its setting configuration window.

Figure 12-13. Example of IP tagging policy



3 In the setting configuration window, specify values for the following parameters:

- IPTag

  For example: USCASJ-ERX-VZ

  - MgmtIPAddrOrRange

    For example: 192.168.35.<1-2>

  - MonitorMgmtIP

    For example: MANAGED

  - MonitorOverlappingIP

    For example: UNMANAGED

  - OverlappingIPAddrOrRange

    For example: 192.168.<35-45>.*|192.168.100.<10-20>

  - RouteDistinguisher

    For example: ENABLE_RD

4 Click **Apply** to save your changes.

5 In the tagging group tree, click **IP Tag Filter-1** to display the configuration window.

6 In the configuration window, click **Matching Criteria** to display the **Matching Criteria** tab.

7 In the **Matching Criteria** tab, specify the additional matching criteria that will allow only the routers in the target customer network to become members of this group. The procedure for specifying matching criteria is given in Editing matching criteria.

Note Specify matching criteria broad enough so that *all* routers in the target network become members of this group. Otherwise, the IP Manager will create multiple names (multiple IPNetwork instances) for the target network, which will produce incorrect correlation analysis results.

For example, if two distinguishing features for the routers in the target customer network is that they are all routers having names beginning with the string “net1,” set the **Type** attribute to **Router** and the **Name** attribute to **net1*** to ensure that only those routers become members of the IP Tag Filter-1 group.

For any router that becomes a member of this group, the names of all IP objects created for the router IP addresses that fall within the OverlappingIPAddrOrRange range that is specified in step 3 will be tagged with the IPTag value that you specified in step 3.

8 Click **Apply** to save your changes.

9 Click **Reconfigure** to implement your changes.

To create additional system IP tag groups for systems, create new groups from the group configured in the preceding procedure and repeat that procedure for the newly created groups. The procedure for creating new groups is given in Creating new groups.

Create an IP tag filter for each group of systems that are using the same, but locally distinct IP addresses.

Verifying IP tag group configurations

To verify that your system IP tag groups are configured properly:

1 Attach the Domain Manager Administration Console to the target IP Manager.

Instructions for opening the Domain Manager Administration Console are presented in Opening the Domain Manager Administration Console.

2 In the console, select **Topology** > **Discover All** to rediscover all managed systems.

3 In the topology tree of the console, expand the IP class to see the tagged IP instances (and their associated systems) that the IP Manager just created.

4 Verify that the rediscovered IPs are associated with the appropriate systems.

Audit Logging

13

This chapter includes the following topics:

- Audit Logging format
- Installing the Audit Log Adapter
- Configuring audit logging
- Starting the Audit Log Adapter
- These two files are located in the local/logs directory relative to the adapter. So, if the adapter is installed in a separate base directory than the server it is auditing, the adapter will read the raw audit log file from the server local/logs directory and write the final audit log file to the adapter local/logs directory.Configuring audit logging
- Starting the Audit Log Adapter
- Domain Manager Administrator Console user actions

Audit Logging format

Audit log contains one line per entry. The entry consists of multiple tab-delimited fields. The first N fields are of fixed meaning and order, with additional fields in a “tag=value” format. The tagged fields that are present depend on the action that is logged. Different actions will have different additional data available. For example, for an “invoke”, the tag “oper” would indicate the operation being invoked.

The order of the tab-delimited fields in an entry is:

```
date login ID ClientDescription pid IP user@host action [tag1=value [tag2=value […]]]
```

Example:

```
2010/03/25 12:52:21 +442ms  admin  2  dmctl   23168  127.0.0.1  root@host01.example.com
createInstance    object=MyClass::MyInstance
```

The meaning of the fixed fields:

- Date — The date and time at which the event occurred, formatted according to the setting of SM_DATETIME_FORMAT.

- ID — The client identification number for this session.
- Login — The login identifier used to authenticate to the domain.
- Client Description — Information on the kind of client, eg dmctl, console, etc. This information is self-reported by the client as it cannot independently verified. A malicious user could conceivably forge this information.
- PID — The process ID of the client on the system where the connection originated. This information is self-reported by the client as it cannot be independently verified. A malicious user could conceivably forge this information.
- IP — The network address of the system from which the connection originated. This is obtained from the network stack. Due to the presence of NAT or other factors it may not reflect the actual IP of the client system.
- User — The OS login name of the user on the system where the connection originated. This information is self-reported by the client as it cannot be independently verified. A malicious user could conceivably forge this information.
- Host — The hostname of the system where the connection originated. This name is self-reported by the client and is subject to forgery. If there is any doubt, perform a reverse lookup on the IP provided instead. This field is provided as it may allow identification of individual clients behind a NAT, while a reverse lookup of the IP would resolve to a single router.
- Action — The API operation called by the client. Most commonly, “put”, “create”, “delete”, or “invoke”.

  Actions defined include the following. It is possible new actions may be added in the future. Many actions are read-only and thus would not appear in the audit log normally, and so are not listed here. Not all of these will be commonly seen.

- consistencyUpdate — Requests that the codebook be recomputed.
- correlate — Causes the domain manager to begin a correlation.
- create — Create a new object instance in the domain. Tagged values include “object”, the class name and instance name of the object to create.
- delete — Delete the object with the given name. Tagged values include “object”, the object to be deleted.
- exec — Causes the domain to executes a program object (.po). Tagged values include “program”.
- insert — Add a new value to a table of an object. Tagged values include “attribute”, which identifies the object and attribute to be affected, and “value”, the desired new value.
- invoke — Invoke a model operation. If an operation can be determined to be read-only, it will not be logged. Tagged values include “object”, the name of the object on which to invoke the operation, “oper”, the name of the operation, and “args”, the list of argument values.
- loadLibrary — Requests that the domain manager load a shared library, such as one containing a compiled model. Tagged values include “library”.

- loadProgram — Request to load a program object (.po) into the domain. Tagged values include “program”
- put — Set an attribute of an object. Tagged values include “attribute”, which identifies the object and attribute to be affected, and “value”, the desired new value.
- remove — Remove a value from a table of an object. Tagged values include “attribute”, which identifies the object and attribute to be affected, and “value”, the value to be removed.
- removeByKey — Remove a value from a table of an object. Tagged values include “attribute”, which identifies the object and attribute to be affected, and “key”, the key of the value to be removed.
- restore — Restore object instances from a file. Tagged values include “filename”, the file to restore from.
- save — Store the repository to a file. Tagged values include “filename”, the file the objects will be stored in, and “class” if the request was to save a subset of instances by class.
- shutdown — Requests that the domain manager shut down.
- transactionAbort — Abort a previously started repository transaction.
- transactionCommit — Commit a previously started repository transaction
- transactionStart — Start a repository transaction.

Tagged Value

Tagged values that may appear include:

- args — A list of values used as argument to an operation.
- attribute — The fully-qualified name of an attribute of an object, in the format “ClassName::InstanceName::AttributeName “
- class — The name of a model class.
- filename — The name of a file (whose use will be clear from the action)
- value — A value used as a table key.
- library — The name of a shared library
- object — The name of an object instance, in the format “ClassName::InstanceName”
- oper — The name of a model operation.
- program — The name of a program object (.po) loaded or executed
- value — The value of an attribute.

Installing the Audit Log Adapter

The audit log adapter is included with the IP Manager. Chapter 3, Installing IP Management Suite, in the VMware Smart Assurance Installation Guide for SAM, IP, ESM, MPLS, and NPM Managers provides information related to the installation of the audit log adapter.

Configuring audit logging

To generate audit log output on a server, you must perform the following steps.

1 Add the *SM_AUDIT_CLIENT_ACTIONS* environment variable to the *runcmd_env.sh* file located in the *BASEDIR*/*<productsuite>*/smarts/local/conf directoryof your VMware Smart Assurance software product suite.

   For example, the directory for the IP AM/PM suite would be *BASEDIR*/IP/smarts/local/conf.

2 You can do this by using the *sm_edit* utility via the command line to open the *runcmd_env.sh* file. Invoke *sm_edit* from the *BASEDIR*/*<productsuite>*/smarts/bin directory:

```
sm_edit conf/runcmd_env.sh

#/opt/InCharge/IP/smarts/bin>sm_edit conf/runcmd_env.sh
```

3 Add the environment variable and its value using the following syntax:

```
SM_AUDIT_CLIENT_ACTIONS=TRUE
SM_AUDIT_ALL_CONNECTIONS=1
```

4 Save the *runcmd_env.sh* file and close it.

5 Restart any VMware Smart Assurance program currently running for the new environment variables to take effect.

Starting the Audit Log Adapter

There are two configuration files involved in starting the VMware Smart Assurance Audit Log Adapter. These files are located in the *BASEDIR /<productsuite>/smarts/*local/conf/audit_parse directory.

The two configuration files are:

- *audit_table.conf*—contains data tables that control the output of the audit log adapter. These tables indicate how to process different entries.
- *adapter_name.conf—specifies the location of the raw audit log file to be parsed. The* BaseSmartsDir *and* AuditInputFile *parameters in this file must be modified to provide the location of the raw audit log file <server_name>_audit.log. You can edit* the *adapter_name.conf* file by using the *sm_edit* utility.

  The BaseSmartsDir parameter specifies the base directory path for VMware Smart Assurance software (for example, /opt/InCharge/IP/smarts/).

  The AuditInputFile parameter specifies the correct log file from which to parse output. This value must be relative to the BaseSmartsDir parameter above, (for example, local/logs/INCHARGE-AM_audit.log).

  There are also three ASL files involved in invoking the Audit Log Adapter. These files are located in the *BASEDIR /<productsuite>/smarts/local/rules/audit_parse* directory. The three ASL files are:

- *parse_audit_log.asl*—parses the audit log output from the server and creates the final audit log file using the information in the *audit_table.conf* file.
- *import_audit_tables.asl*—reads the *audit_table.conf* file.
- *import_audit_params.asl—reads the adapter_name.conf file.*

  To start the Audit Log Adapter, you must perform the following steps:

  a Copy and rename the *adapter_name.conf* file in the *BASEDIR /<productsuite>/smarts/*local/conf/audit_parse directory. Name the new file *<adapter_name>.conf so that it matches the <adapter_name> given in the startup command syntax in* step 3 on page 128 *. For example, if you are going to start up the INCHARGE-AM-AUDIT adapter in* step 3 *, then you would copy and rename the adapter_name.conf* file to *INCHARGE-AM-AUDIT.conf.*

  Then you must edit this file to contain the appropriate BaseSmartsDir and AuditInputFileparameters *to provide the correct location of the raw audit log file to be parsed*. The BaseSmartsDir and AuditInputFileparameters are described above in the description of the *adapter_name.conf file*.

  Note To have the same adapter installation run two separate adapters connecting to two different raw audit logs, you can create two separate *<adapter_name>.conf* files, each specifying a different raw audit log file. This enables you to start two different adapter processes with different adapter names.

  b Before the Audit Log Adapter is started, there must be an existing raw audit log file for the adapter to parse. A raw audit log file is only created when the server has been started, and a Global Console attaches. In order to avoid this dependency on the startup order, you should create an empty raw audit log file prior to starting up the adapter. You can do this by performing the following commands:

  c cd *<base smarts dir>*

  where:

  <base smarts dir> is the base VMware Smart Assurance directory where the server is running

  d cd local/logs

  e touch <server_name>_audit.log

  f Start the Audit Log Adapter from the command line using the following command:

  t *BASEDIR*/*<product_suite>*/smarts/bin/sm_server -n *<adapter_name>* --config=audit_parse --output --daemon s

  where:

  <product_suite> represents the VMware Smart Assurance product suite that the product is part of. For example, IP AM is part of the VMware Smart Assurance IP product suite in the */opt/InCharge /IP* directory.

<adapter_name> is the name of the Audit Log Adapter. It is recommended that you name the Audit Log Adapter relative to the server for which it is doing audit parsing, for example INCHARGE-AM-AUDIT.

Note ts Indicates the command must be typed as one line.

After the adapter is invoked, it runs as a server.

There are two output files associated with the adapter:

- *<adapter_name>.log*—the output file from running the adapter.
- *<server>_audit.log-final*—the final audit log output.

These two files are located in the *local/logs* directory relative to the adapter. So, if the adapter is installed in a separate base directory than the server it is auditing, the adapter will read the raw audit log file from the server *local/logs* directory and write the final audit log file to the adapter *local/logs* directory.Configuring audit logging

To generate audit log output on a server, you must perform the following steps.

1 Add the *SM_AUDIT_CLIENT_ACTIONS* environment variable to the *runcmd_env.sh* file located in the *BASEDIR/<productsuite>/smarts/local/conf* directory of your VMware Smart Assurance software product suite.

   For example, the directory for the IP AM/PM suite would be *BASEDIR/IP/smarts/local/conf*.

2 You can do this by using the *sm_edit* utility via the command line to open the *runcmd_env.sh* file. Invoke *sm_edit* from the *BASEDIR/<productsuite>/smarts/bin* directory:

```
sm_edit conf/runcmd_env.sh
#/opt/InCharge/IP/smarts/bin>sm_edit conf/runcmd_env.sh
```

3 Add the environment variable and its value using the following syntax:

```
SM_AUDIT_CLIENT_ACTIONS=TRUE
SM_AUDIT_ALL_CONNECTIONS=1
```

4 Save the *runcmd_env.sh* file and close it.

5 Restart any VMware Smart Assurance program currently running for the new environment variables to take effect.

Starting the Audit Log Adapter

There are two configuration files involved in starting the VMware Smart Assurance Audit Log Adapter. These files are located in the BASEDIR/<productsuite>/smarts/local/conf/audit_parse directory.

The two configuration files are:

- audit_table.conf—contains data tables that control the output of the audit log adapter. These tables indicate how to process different entries.
- adapter_name.conf—specifies the location of the raw audit log file to be parsed. The BaseSmartsDir and AuditInputFile parameters in this file must be modified to provide the location of the raw audit log file <server_name>_audit.log. You can edit the adapter_name.conf file by using the sm_edit utility.

  The BaseSmartsDir parameter specifies the base directory path for VMware Smart Assurance software (for example, /opt/InCharge/IP/smarts/).

  The AuditInputFile parameter specifies the correct log file from which to parse output. This value must be relative to the BaseSmartsDir parameter above, (for example, local/logs/INCHARGE-AM_audit.log).

  There are also three ASL files involved in invoking the Audit Log Adapter. These files are located in the BASEDIR/<productsuite>/smarts/local/rules/audit_parse directory. The three ASL files are:

- parse_audit_log.asl—parses the audit log output from the server and creates the final audit log file using the information in the audit_table.conf file.
- import_audit_tables.asl—reads the audit_table.conf file.
- import_audit_params.asl—reads the adapter_name.conf file.

  To start the Audit Log Adapter, you must perform the following steps:

  a Before the Audit Log Adapter is started, there must be an existing raw audit log file for the adapter to parse. A raw audit log file is only created when the server has been started, and a Global Console attaches. In order to avoid this dependency on the startup order, you should create an empty raw audit log file prior to starting up the adapter. You can do this by performing the following commands:

  b cd <base smarts dir>

  where:

  <base smarts dir> is the base VMware Smart Assurance directory where the server is running

  c cd local/logs

  d touch <server_name>_audit.log

  e Copy the *adapter_name.conf* file from *BASEDIR/<productsuite>/smarts/conf/audit_parse* directory to *BASEDIR/<productsuite>/smarts/local/conf/audit_parse* directory. Rename the *adapter_name.conf* file to *<adapter_name>.conf* .

  For example, if you are going to start up the INCHARGE-AM-AUDIT adapter, then you would copy and rename the *adapter_name.conf* file to *INCHARGE-AM-AUDIT.conf*.

f Edit the *<adapter_name>.conf* fileto contain the appropriate BaseSmartsDir and AuditInputFile parameters to provide the correct location of the raw audit log file to be parsed. The BaseSmartsDir and AuditInputFile parameters are described above in the description of the adapter_name.conf file.

Also edit the path in the file to point to the location of the log file, *<server_name>_audit.log*.

Note To have the same adapter installation run two separate adapters connecting to two different raw audit logs, you can create two separate *<adapter_name>.conf* files, each specifying a different raw audit log file. This enables you to start two different adapter processes with different adapter names.

g Edit the *bootstrap.conf* file to comment out the Ruleset as follows:

```
#RuleSet = "audit-parse/init.asl"
```

h Start the Audit Log Adapter from the command line using the following command:

```
t BASEDIR/<product_suite>/smarts/bin/sm_server -n <adapter_name> --config=audit_parse --output
--daemon s
```

After the adapter is invoked, it runs as a server.

There are two output files associated with the adapter:

- <adapter_name>.log.the output file from running the adapter.
- <server>_audit.log-final.the final audit log output.

These two files are located in the local/logs directory relative to the adapter. So, if the adapter is installed in a separate base directory than the server it is auditing, the adapter will read the raw audit log file from the server local/logs directory and write the final audit log file to the adapter local/logs directory.

Domain Manager Administrator Console user actions

This section lists some of the common tasks that a user may perform in the Domain Manager Administration Console, and that are captured in audit logs. This list is not intended to be an exhaustive list, but rather a representative sample of some of the more frequently-used modifications in the Domain Manager.

Polling and thresholds

The following actions relate to polling and threshold settings. These actions can be performed by selecting **Edit**, **Polling and Thresholds** in the Domain Manager Administration Console, selecting the appropriate device class/groups in the left tree pane, and then selecting the appropriate Settings, Priorities, and Matching Criteria tabs in the right pane.

- Modify selection criteria

- Modify group priority
- Add a new setting
- Remove a setting
- Reconfigure
- Restore defaults (for a setting)

Configuring groups

The following actions relate to the configuration of groups. These actions can be performed by selecting **Configure**, **Topology Builder** in the Domain Manager Administration Console, and then by selecting **Configure**, **Groups** in the Topology Builder.

- New group
- Delete group

Topology-related changes

The following actions relate to topology changes.

The following actions can be performed by clicking an InCharge Manager listed in the left pane of the Domain Manager Administration Console (for example, INCHARGE-IPAMPM), clicking the Topology tab in the right pane, and choosing options in this view.

- Change discovery settings
  - Discover pending interval
  - Full discovery interval
  - Enable/disable auto discovery
  - Change auto discovery types
  - Community string list add
  - Community string list delete
  - Change system limit

    The following actions can be performed by clicking an InCharge Manager listed in the left pane of the Domain Manager Administration Console (for example, INCHARGE-IPAMPM), clicking the Discovery Filters tab in the right pane, and choosing options in this view.
- Discovery filters
  - Add a filter
  - Modify a filter
  - Delete a filter
  - Change filter priority order

Discovery actions

The following actions relate to discovery. These actions can be performed by selecting *Topology* in the Domain Manager Administration Console and selecting the following menu items.

- Import from seed file
- Add Agent
- Rediscover
- Discover Pending
- Discover All
- Save Topology

  You can also perform the following actions in the Pending Devices List that displays when you select *Show Discovery Progress* from the *Topology* menu of the Domain Manager Administration Console.

- Mark Non-SNMP
- Mark SNMP only
- Remove from pending list
- Throttle (accept or reject)
- Stop auto discovery

Note These options are displayed by selecting a new device row and right-clicking on it.

Correlation-related changes

The following actions relate to correlation changes. The Recompute Codebook and Correlate Now actions can be performed by selecting **InCharge Manager**, **Recompute Codebook**, or **Correlate Now** in the Domain Manager Administration Console. You can change the correlation interval change by selecting the InCharge Manager listed in the left tree pane (for example, INCHARGE-IPAMPM), and clicking the Correlation tab.

- Recompute codebook
- Correlate now
- Correlation interval change

Changes to devices

The following actions relate to device changes and can be performed by selecting a device in the left tree pane of the Domain Manager Administration Console and selecting *Edit*, or by right-clicking on the device.

- Delete

- Manage a device
- Unmanage a device

Name Resolution

14

This chapter includes the following topics:

- Overview
- Configuring automatic name resolution
- Configuring exceptions to automatic name resolution
- Configuring seed-name name resolution
- Configuring exceptions to seed-name name resolution
- Three name resolution — Related tasks
- Local exceptions to automatic and seed-file name resolution

Overview

The discovery process uses the name resolution service that is provided by the local operating system to translate hostnames to IP addresses, and to translate IP addresses to hostnames. On a UNIX system, a name resolution service might be Domain Name System (DNS), NIS or NIS+, or a hosts file.

This chapter describes the following tasks related to name resolution:

- *"Configuring automatic name resolution" on page 152*
- Configuring exceptions to automatic name resolution
- Configuring seed-name name resolution
- Configuring exceptions to seed-name name resolution
- Three name resolution — Related tasks
- Local exceptions to automatic and seed-file name resolution

Configuring automatic name resolution

The TM_USEAUTONAME option in the name-resolver.conf file enables you to configure the Name Resolution probe to automatically resolve system names by using the naming sources in the order that you specify.

- “Configuring automatic name resolution” section in Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides more information on naming-source sequence used by the Name Resolution probe to resolve system names.
- “Configuring automatic name resolution”section in Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides more information on AutoNameOrder naming sources.

  To configure automatic name resolution:

  a Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the name-resolver.conf file:

  ```
  sm_edit conf/discovery/name-resolver.conf
  ```

  b Find the **NameFormat** section of the file:

  ```
  #NameFormat = "TM_USESEEDNAME"
  NameFormat = "TM_USEAUTONAME"
  ```

  c Ensure that the **NameFormat = “TM_USEAUTONAME”** option is the only one that is *not* commented out.

  d Find the **AutoNameOrder** section of the file:

  ```
  AutoNameOrder 0 TM_USELOOPBACK
  AutoNameOrder 1 TM_USESYSNAME
  AutoNameOrder 2 TM_USENONPRIVATEIP
  AutoNameOrder 3 TM_USEPRIVATEIP
  AutoNameOrder 4 TM_USEAGENTADDRESS
  ```

  e Change the **AutoNameOrder** numbers (0 through 4) to the order that you prefer. The resolution process is performed in ascending order, starting with **AutoNameOrder 0**. AutoNameOrder naming sources describes the **AutoNameOrder** naming sources.

Table 14-1. AutoNameOrder naming sources

Naming source	Description
TM_USELOOPBACK	Resolve loopback IP address: If a loopback interface IP address can be resolved to a name, that name will be used to name the system. By default, the default loopback interface type number is 24, which represents the “software loopback” interface type. For a discovered system that has multiple loopback addresses, the Name Resolution probe tries to resolve the lowest loopback address first. If that resolution fails, the Name Resolution probe tries to resolve each of the remaining loopback addresses.
TM_USESYSNAME	Resolve MIB-II sysName: If the sysName can be resolved to at least one IP address, and at least one of the resolved IP addresses resolves back to the sysName, the sysName will be used to name the system.
TM_USENONPRIVATEIP	Resolve non-private IP address: If at least one of the IP addresses is not a private IP address, the non-private IP address will be used to resolve the name. Once a name is resolved, the name will be used to name the system.
TM_USEPRIVATEIP	Resolve private IP address: If at least one of the IP addresses is a private IP address, the private IP address will be used to resolve the name. Once a name is resolved, the name will be used to name the system. By default, the private IPv4 address pattern is 10.*|192.168.*|172.<16-31>.*, which is defined as follows: ■ Any IP address with the value of 10 in the first octet (10.*.*.*). ■ Any IP address with a value of 172 in the first octet and a value between 16 and 31 in the second octet (172.<16-31>.*.*). ■ Any IP address with a value of 192 in the first octet and a value of 168 in the second octet (192.168.*.*).
TM_USEAGENTADDRESS	Resolve SNMP agent address: If an SNMP agent address can be resolved to a name, that name will be used to name the system.

1 Save and close the file.

   The modified version of the name-resolver.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

2 Type the following command from BASEDIR/smarts/bin to implement your changes:

```
            sm_tpmgr -s <
               IP Manager

             instance name> -b <
            brokerName:port
            >

--load-conf=name-resolver.conf
            sm_tpmgr -s INCHARGE-AM -b localhost:426  --load-conf=name-resolver.conf
```

3 Optional: Configure exceptions to automatic name resolution by using a seed file, as described in *“Configuring exceptions to automatic name resolution” on page 154*.

   If the resolution process is unsuccessful for all naming sources that are defined in the name-resolver.conf file, a final procedure uses a public IP address, a private IP address, or the SNMP agent address for the system name.

Configuring exceptions to automatic name resolution

You can define local exceptions to automatic name resolution by adding the keywords and values that are identified in the left column of Configuring exceptions to automatic name resolution to specific systems in the seed file. Consult How to create a seed file.

Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides information on the format of a seed file.

To configure exceptions for automatic name resolution:

1 Configure automatic name resolution as described in *“Configuring automatic name resolution” on page 152*.

2 Create a seed file and, for each exception to the sources that are defined for automatic name resolution, add an entry for the system with either a specific naming source or a specific name:

   - To define a specific naming source for a system, add the following keyword and value to the seed file entry:

```
NAMEFORMAT=<naming source>
192.168.12.2 NAMEFORMAT=TM_USELOOPBACK
192.168.2.2 NAMEFORMAT=TM_USEPRIVATEIP
NAMEFORMAT=TM_USESEEDNAME SEEDNAME=<name>
ROUTER2 NAMEFORMAT=TM_USESEEDNAME SEEDNAME=Core-router-2
192.168.8.29 NAMEFORMAT=TM_USESEEDNAME SEEDNAME=My-router
```

3 Import the seed file for discovery.

   The locally specified naming sources and names will be used to name the systems.

   If the resolution process is unsuccessful for the specified naming source for a system, a final procedure will use a public IP address, a private IP address, or the SNMP agent address for the system name.

Configuring seed-name name resolution

When you select the TM_USESEEDNAME option in the name-resolver.conf file, the Name Resolution probe uses the following keyword and value that are defined for individual systems in the seed file to name the systems:

SEEDNAME=<*user-specified name*>

For any system in the seed file for which this keyword is not defined, the Name Resolution probe uses the seed name, which is the name by which the system was introduced to the discovery process, to name the system. For an autodiscovery candidate system, for which a seed name is not available, the Name Resolution probe uses the first non-private IP address to name the system.

To configure seed-name name resolution:

1 Go to the BASEDIR/smarts/bin directory in the IP Manager installation area and type the following command to open the name-resolver.conf file:

```
sm_edit conf/discovery/name-resolver.conf
```

2 Find the **NameFormat** section of the file:

```
NameFormat = "TM_USESEEDNAME"
#NameFormat = "TM_USEAUTONAME"
```

3 Ensure that the **NameFormat = “TM_USESEEDNAME”** option is the only one that is *not* commented out.

Steps 4 and 5 are optional. Only if you want to use the NAMEFORMAT=TM_USEAUTONAME exception that is described in the right column of Configuring seed-name name resolution do you need to perform these steps.

4 Optional: Find the **AutoNameOrder** section of the file:

```
AutoNameOrder 0 TM_USELOOPBACK
AutoNameOrder 1 TM_USESYSNAME
AutoNameOrder 2 TM_USENONPRIVATEIP
AutoNameOrder 3 TM_USEPRIVATEIP
AutoNameOrder 4 TM_USEAGENTADDRESS
```

5 Optional: Change the **AutoNameOrder** numbers (0 through 4) to the order that you prefer. The resolution process is performed in ascending order, starting with **AutoNameOrder 0**. Configuring seed-name name resolution describes the **AutoNameOrder** naming sources.

6 Save and close the file.

The modified version of the name-resolver.conf file is saved to the BASEDIR/smarts/local/conf/discovery directory.

7 Type the following command from BASEDIR/smarts/bin to implement your changes:

```
sm_tpmgr -s <
   IP Manager

 instance name> -b <
```

```
                brokerName:port
                >  --load-conf=name-resolver.conf
                sm_tpmgr -s INCHARGE-AM -b localhost:426  --load-conf=name-resolver.conf
```

8 Create a seed file and define a specific name for each system by adding the following to each entry:

```
SEEDNAME=<name>
ROUTER2 SEEDNAME=Core-router-2
192.168.8.29 SEEDNAME=My-router
```

9 Optional: Configure exceptions to seed-file name resolution, as described in *"Configuring exceptions to seed-name name resolution" on page 156*.

10 Import the seed file for discovery.

The locally specified names for systems will be used to name the systems.

If the resolution process is unsuccessful for the specified naming source for a system, a final procedure will use a public IP address, a private IP address, or the SNMP agent address for the system name.

Configuring exceptions to seed-name name resolution

You can define local exceptions to seed-name name resolution by adding the keywords and values that are identified in the right column of Configuring exceptions to seed-name name resolution to specific systems in the seed file.Chapter 2 in VMware Smart Assurance IP Manager Concepts Guide describes the format of a seed file.

To configure exceptions for seed-name name resolution:

1 Configure seed-name name resolution as described in *"Configuring seed-name name resolution" on page 155*.

2 Revise your seed file and, for each exception entry in the seed file, add a keyword and value as follows:

```
NAMEFORMAT=<naming source>
 192.168.6.5 NAMEFORMAT=TM_USEAUTONAME
 192.168.12.2 NAMEFORMAT=TM_USELOOPBACK
 192.168.2.2 NAMEFORMAT=TM_USEPRIVATEIP
```

3 Import the seed file for discovery.

The locally specified naming sources will be used to name the systems.

If the resolution process is unsuccessful for the specified naming source for a system, a final procedure will use a public IP address, a private IP address, or the SNMP agent address for the system name.

Three name resolution — Related tasks

The discovery process relies on the local name resolution service to complete the following three tasks:

- For manual discovery, resolve the hostnames of discovery candidate systems to corresponding IP addresses so that the systems can be discovered.
- For manual discovery, resolve the IP addresses of discovered systems to corresponding hostnames so that the discovered systems are given unique names within the topology.
- For autodiscovery, resolve the IP addresses of discovered systems to corresponding hostnames so that:
  - Hostnames can be matched against the “System Name” field in the discovery filters that are described in *“Creating discovery filters” on page 88*.
  - Discovered systems that pass the discovery filters are given unique names within the topology.

    Until a system is discovered, it is known to the discovery process by its IP address only. When the system is discovered, the discovery process makes calls to the local name resolution service and runs a script to assign the system a distinct and meaningful name.

Local exceptions to automatic and seed-file name resolution

For both automatic and seed-file name resolution, you can define local exceptions for individual systems in the seed file. Local name resolution exceptions for individual systems in a seed file identifies the keyword=<value> exceptions.

Table 14-2. Local name resolution exceptions for individual systems in a seed file

Automatic name resolution 1	Seed-name name resolution 2
1 When you specify NameFormat = "TM_USEAUTONAME" in the name-resolver.conf file. 2 When you specify NameFormat = "TM_USESEEDNAME" in the name-resolver.conf file.	
Individual automatic name resolution exception by using a prioritized list of naming sources: Not applicable	Individual automatic name resolution exception by using a prioritized list of naming sources: ■ NAMEFORMAT=TM_USEAUTONAME The resolution process is performed in accordance to the naming source order (0 -> 4) that is specified by the AutoNameOrder entries in the name-resolver.conf file. The default order is: ■ AutoNameOrder 0 TM_USELOOPBACK ■ AutoNameOrder 1 TM_USESYSNAME ■ AutoNameOrder 2 TM_USENONPRIVATEIP ■ AutoNameOrder 3 TM_USEPRIVATEIP ■ AutoNameOrder 4 TM_USEAGENTADDRESS if you prefer a different naming source order, edit the name-resolver.conf file and change the AutoNameOrder numbers.

Table 14-2. Local name resolution exceptions for individual systems in a seed file (Continued)

Automatic name resolution 1	Seed-name name resolution 2
Individual automatic name resolution exceptions by using an individual naming source: ■ NAMEFORMAT=TM_USELOOPBACK ■ NAMEFORMAT=TM_USESYSNAME ■ NAMEFORMAT=TM_USENONPRIVATEIP ■ NAMEFORMAT=TM_USEPRIVATEIP ■ NAMEFORMAT=TM_USEAGENTADDRESS	Individual automatic name resolution exceptions by using an individual naming source: ■ NAMEFORMAT=TM_USELOOPBACK ■ NAMEFORMAT=TM_USESYSNAME ■ NAMEFORMAT=TM_USENONPRIVATEIP ■ NAMEFORMAT=TM_USEPRIVATEIP ■ NAMEFORMAT=TM_USEAGENTADDRESS
Individual seed-file name resolution exception by specifying a specific name: ■ NAMEFORMAT=TM_USESEEDNAME SEEDNAME=<*user-specified name*>	Individual seed-file name resolution exception by specifying a specific name: Not applicable

For an ICMPONLY system, if automatic name resolution is in effect, use the NAMEFORMAT=TM_USESEEDNAME and SEEDNAME=<*user-specified name*> exception to name the system. If seed-file name resolution is in effect, use SEEDNAME=<*user-specified name*> to name the system.

The procedure for defining local exceptions to automatic name resolution is given in Configuring exceptions to automatic name resolution

The procedure for defining local exceptions to seed-name name resolution is given in Configuring exceptions to seed-name name resolution.

Working with topology

15

This chapter includes the following topics:

- Adding read community strings
- Deleting read community strings
- Specifying additional read community strings
- Observing the management state of an object
- Setting the management state of an object
- Managing or unmanaging filesystems
- Topology Change Analysis Tool (TCAT)

Adding read community strings

To add a read community string to the Read Community Strings window:

1 Attach the Domain Manager Administration Console to the target IP Manager.

   Opening the Domain Manager Administration Console provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, click the IP Manager name to display a multiple-tab window in the right panel of the console.

3 Click **Topology**.

4 Type a read community string in the **Community Strings** field.

5 Click **Add**. By default, you can list up to four community strings in the **Read Community Strings** window.

6 Click **Apply**.

   The changes that you make to the Read Community Strings window are applied to systems that are discovered after the changes are made. Systems that are already discovered are not affected.

Deleting read community strings

To delete a read community string from the Read Community String**s** window:

1 Attach the **Domain Manager Administration Console** to the target IP Manager.

Opening the Domain Manager Administration Console provides instructions on opening this console.

2 In the left panel of the **Domain Manager Administration Console**, click the IP Manager name to display a multiple-tab window in the right panel of the console.

3 Click **Topology**.

4 In the **Read Community Strings** window, select the community string that you want to remove.

5 Click **Remove**, and then click **Apply**.

The default read community string *public* is automatically included in the Read Community Strings window. If you do not want the IP Manager to use the *public* community string, you must remove *public* here and from the defaultReadCommunity parameter in the discovery.conf file.

Chapter 6, Configuration Settings in the VMware Smart Assurance IP Manager Reference Guide describes the configuration parameters in the discovery.conf file and provides instructions on modifying the parameters.

Specifying additional read community strings

To specify additional read community strings, you need to change the value of the *MaximumCommunities* parameter in the discovery.conf file.

Observing the management state of an object

To observe the management state of an object:

1 Attach the Domain Manager Administration Console to the target IP Manager.

Opening the Domain Manager Administration Console provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, select the object to display the object's properties window in the right panel of the console.

3 In the object's properties window, click the **Attributes** tab and locate the **IsManaged** attribute.

The value of this attribute indicates the management state of the object.

Setting the management state of an object

You can invoke the **Manage** and **Unmanage** commands through a Global Console that is attached to the IP Manager.

To set the management state of an object by using the **Manage** and **Unmanage** commands:

1 Attach the Domain Manager Administration Console to the target IP Manager.

Opening the Domain Manager Administration Console provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, select the object that is to be managed or unmanaged, right-click, and then select **Manage** or **Unmanage** from the pop-up menu.

3 Click the **Reconfigure** toolbar button.

The **Reconfigure** toolbar buttons are described in Domain Manager Administration Console toolbar buttons.

Note When you manually change the management state of an object, you must reconfigure for the change to take effect.

Figure 15-1. Domain Manager Administration Console toolbar buttons

Button	Description
	Attach to a Manager
	Detach from a Manager
	Import topology from a seed file
	Add a new SNMP agent to the topology
	Add a new source to the topology **Note:** The Add Source icon is deactivated by default.
	Reconfigure: update instrumentation, recompute codebook, reapply polling/threshold values, and so on
	Open the Polling and Thresholds Console

Correcting a duplicate IP address error

To correct a duplicate IP address error:

1 Attach the Domain Manager Administration Console to the target IP Manager.

Opening the Domain Manager Administration Console provides instructions on opening this console.

2 In the left panel of the Domain Manager Administration Console, click the **+** sign to expand the **DuplicateIP** class, select the DuplicateIP object, right-click, and then select **Delete** from the pop-up menu.

3 Use the IP tagging feature to give each of the duplicate IP addresses a unique tag.

4 Rediscover all systems that contain duplicate IP addresses. Consult Starting discovery from the console to complete this step.

Managing or unmanaging filesystems

You can manage or unmanage an individual filesystem instance. When a filesystem is rendered managed, that filesystem instance is monitored. On the other hand, when a filesystem instance is rendered unmanaged, that filesystem instance is not monitored. Follow these steps to manage or unmanage a filesystem instance:

1 In the **Domain Manager Administration Console**, select the appropriate server from the **Manager** drop-down list.

2 Right-click on the filesystem instance and select **Manage** or **Unmanage**, as applicable.

3 If you have chosen to manage a filesystem instance, click **Reconfigure**. This renders the filesystem to be managed.

Related tasks and information

- "Sample File system groups and matching criteria" section in Chapter 7, Polling and Threshold Settings in the VMware Smart Assurance IP Manager Reference Guide provides information on attributes of FileSystem.
- For procedures to migrate FileSystem settings, see *"Upgrade and migration impacts" on page 64*.
- To define thresholds and change priority of a filesystem group, see Defining thresholds for individual filesystem instances.

Topology Change Analysis Tool (TCAT)

The Topology Change Analysis Tool (TCAT) identifies the key topology differences between various versions of IP releases and provides logical explanations based on any feature changes or enhancements. TCAT helps you to identify changes to a set of objects, relationships, and attributes which are related to device certifications for IP Manager upgrades from 8.1.3, 8.1.4, 9.0, and 9.1 releases to 9.2.

Note Unexplained topology differences may be observed if you have not applied the latest patch for a release. This is because there may be certification changes between patch releases.

Installing TCAT

TCAT is installed as "sm_tcat" while installing IP and is located in the <BASEDIR>\smarts\bin directory.

The tool when invoked, compares the topology from your current deployment with the new version of the IP domain, for example, IP 8.1.4 with IP 9.1 or IP 9.1 with IP 9.2. The tool identifies differences in the two topologies and flags a device if the device has been discovered under a different class in each topology or the same device for example, has different instances listed under ComposedOf, ConnectedVia, MemberOf, Underlying, HostsAccessPoints, MemberOf, and InstrumentedBy relationships. The tool also provides an explanation of why a device is discovered differently or why the relationships, attributes, or connections are different from one version of IP to another using the same seedfile.

Both the old and the new IP installations must be available and running at the same time. The tool can then be invoked to attach to both IP servers, collect the necessary topology information from them, carry out the data analysis on the topology differences, and deduce the corresponding logical explanations.

TCAT files

The following TCAT input files are located under the <BASEDIR>\smarts\conf\tcat directory:

IP_x.x.x_TopoRules.xml

For a logical explanation of topology differences, sets of topology rules are imported into the Rule Engine in the form of XML files. These XML files are located in the <BASEDIR>\smarts\conf\tcat directory, for example, C:\InCharge\IP\smarts\conf\tcat\IP_9.2.0_TopoRules.xml. The IP_9.2.0_TopoRules.xml file contains rules for the 9.2 server.

Note Do not modify the .xml files. These files are populated with rules and the tool will produce undesired results if these files are changed.

TopoDiffLists.txt

TopoDiffLists.txt is a configurable flat text file which allows you to select which type of Objects to compare and what relationships and attributes to compare. If you do not want to compare some Object types which comes as default, place # at the beginning of the line where the Object, Relationship, or Attribute is defined.

You can ignore certain Classes in Relationship in order to reduce the size of report. For example, if there is no need to view the VLAN class in the Relationship, add an entry in the flat file as follows:

```
IgnoredInRel|VLAN
```

tcat.conf

tcat.conf file contains TCAT configuration settings and is located under the smarts/conf/tcat directory.

IPVersionMaps.txt

This file provides mapping of IP versions with Foundation build versions along with the rule XML file.

misc.xml

This file feeds miscellaneous information to the TCAT rule engine. Do not modify the misc.xml file.

Running TCAT

The two IP servers can be on the same machine or geographically separate machines, and in the same broker or different brokers. There is a built-in launcher to invoke this tool with the mandatory arguments for the broker host, broker port, and the domain manager name of both IP servers.

Type the following command to run TCAT:

```
sm_tcat –b1 oldBrokerHost:oldBrokerPort –s1 oldIPServerName –b2 newBrokerHost:newBrokerPort –s2
newIPServerName [ –pw ]
```

The optional “-pw” argument is for interactively entering the non-default userid/password of brokers and domain managers.

Progress summary with relevant performance benchmarks are provided on the console.

TCAT output and the report file

The topology difference between IP releases will be identified and the corresponding logical explanation will be deduced by the Rule. The detailed lists of topology differences for the selected objects, relationships, and attributes, together with the corresponding logical explanation will be presented in the form of a Tab-separated text file (one per row) for you to convert to an Excel spreadsheet.

During the execution of this tool, the summary of progress with the relevant performance benchmarking is provided on the console.

The format of the log file is “tcat_YYYYMMDD_hhmmss_ms.log” and is located in the smarts/local/logs directory. The report text file name is “TopoDiff_version1_vs_version2_YYYYMMDD_hhmmss_ms.txt”, for example, TopoDiff_9.1.0_vs_9.2.0_20120905_155414_300.txt and is also located in the smart/local/logs directory. The timestamp is the tool invocation starting time in milliseconds.

Open the “TCatReportConverter.xlsm” file located under smarts/script/tcat/ on a system with Microsoft Excel to use its VBScript-based Macros for the report conversion and post-processing.

Note Explanations show up in relationships only when there are differences in the OBJ as well. Otherwise, the explanations or reasons are empty.

TCAT terminologies

Naming - TCAT checks if devices in two domains have the same SNMP address and are of the same object type but with different names (case sensitive). If so, the reason is “Naming”.

Derived - For the component Classes, if there is no TopoRule matched, TCAT finds out their underlying Device and derives the reason and automated explanations from there.

Only in one domain (IP 8.1.3, IP 9.0, and so on) - TCAT checks both the IP servers to see whether a device is only in one domain. If so, TCAT flags it to be present in only one domain (the matched TopoRules will not be used) and all associated component classes and relationships will be excluded from the report (this occurs during a later stage of report generation). The topo differences are still identified and stored in a certain data structure but not printed in the report.

Report post processing

The report file is generated as a Tab-separated text file. The report file needs to be converted into an Excel spreadsheet (Microsoft Excel 2003, 2007, or 2010 versions are currently supported) and formatted with various filtering Lists and Pivot Tables to get both the top-level summary views (Slice-and-Dice) and detailed topology difference (Drilldown). A VBScript based Macro automates the entire post-processing of the TCAT report. The Macro carries out all the report formatting, list filtering, and Pivot Table creations.

1 Open the Macro-embedded TCAT Report Converter, that is, the TCatReportConverter.xlsm file available under <BASEDIR>/smarts/script/tcat/.

2 Click “Enable Macros” to open the “Instruction” sheet.

3 Click “File -> Open”, select the report text file, and then click “Finish”.

4 Press Ctrl, Shift and C keys simultaneously. The VBScript based Macro carries out all the report formatting, list filtering, and Pivot Table creations.

5 Save it as an Excel Workbook by “File -> Save As -> Excel Workbook” with the default name.

Note Do not insert or delete any row in the report file because the programmatically generated hyperlinks in the WorkSheet “Report” rely on pre-calculated locations. For example, you can click Cell E2 (merged E2-I2) to jump to the Summary and Statistics section. You may also click Cell A2 (merged A2-D2) anytime (when not filtering) to scroll the sheet to the top (this is important for subsequent filtering because Excel may not display filtering properly if the worksheet is not at the top position).

You can double click on the Excel report output after formatting. That will bring up only those differences on a new sheet for you to drill down on the differences. Clicking on some fields in the Excel output file provides you with the drill down capability. Only the statistic cells in Columns C to F of the first three pivot tables (Summary, BySysOID, and ByDevice) and in Columns E to H of the ByReason pivot table can be clicked for the drill downs in which the detailed information for those selected entries will be pulled from the raw report and placed in a new worksheet.

You can use filters to go over differences one by one and uncheck the differences verified. This will remove the differences from the view enabling you to not look at already verified differences. Also, you can use filters to see only one set of differences based on class, reason, and so on.

Network Attached Storage

16

This chapter includes the following topics:

- Overview
- Installing and starting the HTTPS Adapter service
- Configuring Celerra devices for NAS Extension functionality
- Configuring IP Availability Manager for NAS autodiscovery
- Viewing NAS notifications
- Viewing NAS service topology in maps
- Opening NAS containment

Overview

The NAS Extension is installed with the IP Availability Manager. After installation, the Availability Manager can discover NAS devices, and perform root cause and impact analysis on these devices.

Configuring the NAS Extension involves the tasks summarized in Steps for configuring the NAS Extension.

Table 16-1. Steps for configuring the NAS Extension

	Reference
If necessary, configure the HTTPS Adapter to support access to the managed Celerra devices.	For information about this requirement, refer to the Configuring Control Station usernames and passwords.
If necessary, configure external Control Station and Data Mover IP addresses.	For information about this requirement, refer to the Configuring Control Station and Data Mover IP addresses , which also refers you to the appropriate VMware documentation, if needed.
Install the HTTPS Adapter as a service and start the service.	*"Installing and starting the HTTPS Adapter service" on page 168*
Start the Availability Manager.	*VMware Smart Assurance Installation Guide for SAM, IP, ESM, MPLS, and NPM Managers*

Table 16-1. Steps for configuring the NAS Extension (Continued)

	Reference
Start the Global Manager.	VMware Smart Assurance Installation Guide for SAM, IP, ESM, MPLS, and NPM Managers. For configuration information, refer to the VMware Smart Assurance System Administration Guide.
Start the Global Console.	VMware Smart Assurance Installation Guide for SAM, IP, ESM, MPLS, and NPM Managers.

Installing and starting the HTTPS Adapter service

The NAS Extension software includes an adapter process (the HTTPS Adapter), which probes the Celerra devices using the HTTPS/XML probe to obtain internal topology information. While the installation of this adapter is automatic, you must install the service and start it manually. Install the HTTPS Adapter as a service and start the service manually, as described next for UNIX.

Once started, the HTTPS Adapter registers with the Broker. The VMware Smart Assurance System Administration Guide provides more information about starting services.

UNIX:

To install the HTTPS Adapter as a service on UNIX, issue the following command:

```
# sm_service install --force --unmanaged --startmode=manual \
'--name=<service_name>' \
'--description=<Smarts description>' \
'BASEDIR/smarts/bin/sm_adapter_java'
'--name=<HTTPS_Adapter_Name>' \
'--output=<HTTPS_Adapter_Name>.log' \
'-J' \
'nas_probe.jar'
```

<*Smarts description*> is the description of the service. You may create any description you wish; for example, Remote NAS adapter.

<*HTTPS_Adapter_Name*> is the name of the HTTPS Adapter, as registered with the Broker; for example, the default name is remote-nas.

You can also specify the Broker and Port, if the Availability Manager with NAS Extension is registered with a Broker and Port other than the default, localhost:426. To do this, add the following arguments to the end of the command:

```
--broker=<IP Address or Hostname>:<Port Number>
```

To start the service, type the following command:

```
# BASEDIR/smarts/bin/sm_service start
         <service_name>
```

To stop the service, issue the following command:

```
# BASEDIR/smarts/bin/sm_service stop
         <service_name>
```

HTTPS/XML discovery probe

The HTTPS Adapter probe performs the following tasks:

- Instantiates NASChassis objects (NAS Network Servers)
- Assigns attributes such as software version to the FileServer and NASChassis objects

Configuring Celerra devices for NAS Extension functionality

To support NAS Extension functionality, you may need to configure the following:

- Control Station and Data Mover IP addresses
- Control Station usernames and passwords

Note For NAS devices to be monitored correctly, ensure that SNMP is enabled on the host. You require only read access and not full access to the Control Station for monitoring.

Configuring Control Station and Data Mover IP addresses

The IP Availability Manager Extension for NAS requires access to the Control Station and Data Mover IP addresses that are exposed on the external network. You configure the IP addresses after the devices are cabled to the network switches.

For information about cabling the Celerra Network Server, refer to the Celerra NS500/NS600/NS700/NS704 Integrated Configuration Version 5.4 Setup Guide. Refer to the Connect Cables section, step 5.

The Configuring and Managing Celerra Networking and Configuring and Managing Celerra Network High Availability technical modules provide more information about how to configure the external IP addresses once the cables are in place.

Configuring Control Station usernames and passwords

When using the HTTPS Adapter, the Availability Manager accesses each Celerra Control Station device. Some Control Station devices may restrict access to configured usernames and passwords. If the default username and password (nasadmin) are used, no configuration is required.

Configuring device logins

To configure a login into the Celerra Control Station, issue the following line in the *clientConnect.conf* file:

```
nasUser:<device_id>:<login_id>:<password>
```

where,

- <*device_id*> is either a device name or its IP address
- <*login_id*> is the user name; default username is nasadmin
- <*password*> is the password; default password is nasadmin

Note If an asterisk is used as a wildcard in the final entry of the first field before the nasUser section, that entry will become the default nasUser value. Thus, if you include asterisks in the *clientConnect.conf* file, you must move the *nasUser* section above these entries. This wildcard priority also holds true for definitions within the *nasUser* section; for example, if you configure logins for more than one Celerra Control Station device, ensure that the first login does not contain a wildcard.

“Core fields” section in Chapter 3, Discovery, in the VMware Smart Assurance IP Manager Concepts Guide provides more information on using wildcards.

Adding/removing/modifying logins

You can add/remove/modify these entries anytime using the *sm_edit* utility. In the next discovery process, the modified entries will be active.

Configuring IP Availability Manager for NAS autodiscovery

Enable autodiscovery and create one or more discovery filters to discover the following types of devices in the managed storage environment:

- Host systems (HostSystems)
- Fibre Channel switches (FCSwitches)
- Celerra Control Stations (NAS_ControlStations)
- Celerra Data Movers (NAS_FileServers)

- NAS clients (NASClient_Hosts)

  You create an IP address discovery filter that has the specific values for your NAS devices in the first, second, and third octet positions, and an asterisk (wildcard character) in the fourth octet position; for example, *172.32.14.**.

  The specific values limit the discovery filter to a specific network (in this example, network 172.32.14.0), and the asterisk allows IP Availability Manager to discover all candidate devices belonging to the 172.32.14.0 network. You create an IP-address discovery filter for each network that you want IP Availability Manager to sweep for candidate NAS devices.

  How to create a discovery filter describes the procedure to create discovery filters.

Viewing NAS notifications

You view the notifications, maps, and containment information by attaching the Global Console to the Global Manager. The IP Availability Manager reports NAS device notifications to the Global Manager. You can then view the notifications through the attached Global Console in two basic ways:

- As table entries in a Notification Log Console view
- As color-coded severity icons in a Map Console view

Opening a notification properties dialog box

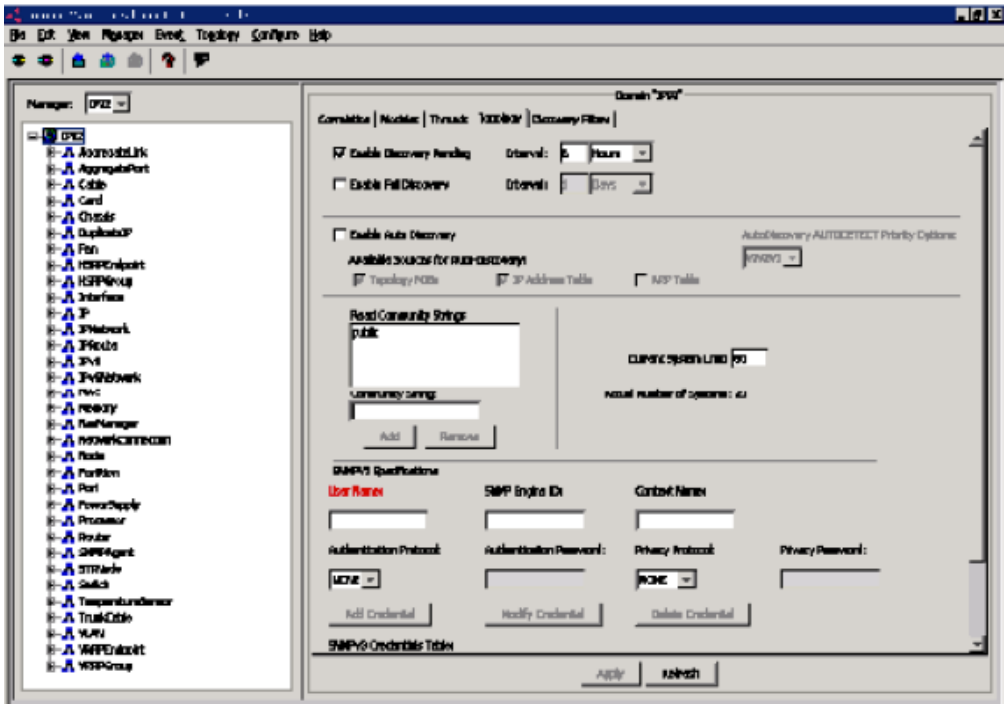
To obtain detailed information about an individual NAS device notification, you can use any of the following common methods to open the Notification Properties dialog box:

- Double-click a NAS device notification in the Notification Log Console.
- Select a NAS device notification in the Notification Log Console and click the **Properties** toolbar button.
- Right-click a selected NAS device notification and select **Properties** in the pop-up menu.
- Double-click a NAS device map icon affected by active events.

NAS device notification properties

Notification Properties dialog box showing fileserver down shows an example of the detailed information displayed in the Notification Properties dialog box for a FileServer Down notification.

Figure 16-1. Notification Properties dialog box showing fileserver down



Viewing NAS service topology in maps

Viewing topology maps is an easy and quick way to learn more about the source, impact, and cause of NAS device notifications. You view the NAS device topology maps using the Map Console view of the Global Console.

Opening a topology map

You can use any of the following common methods to open a NAS device topology map:

- Open the Map Console by selecting the Show Map option from any opened console attached to the Global Manager. For example, in the Notification Log Console, click a NAS device notification and then select **Event > Show Map**, or right-click the notification and then select **Show Map** in the pop-up menu. In the Topology Browser Console, right-click a NAS device element and select **Show Map** in the pop-up menu.
- Open the Map Console from the Global Console by selecting **File > New > Map Console**. In the Topology tab of the Map Console, click NAS element to display a map for the element, or right-click the NAS element and select a map type from the pop-up menu.
- In an opened Topology map, right-click NAS device element map icon and select a map type from the pop-up menu.

Opening NAS containment

The Containment view shows topology information for each type of NAS element. An element's Containment view displays in a dialog box that organizes the information for the element into tab pages and tables.

In a Containment view, different information is available for different types of elements. The containment information is generated by the IP Availability Manager upon request by the Global Manager.

Opening a NAS device containment dialog

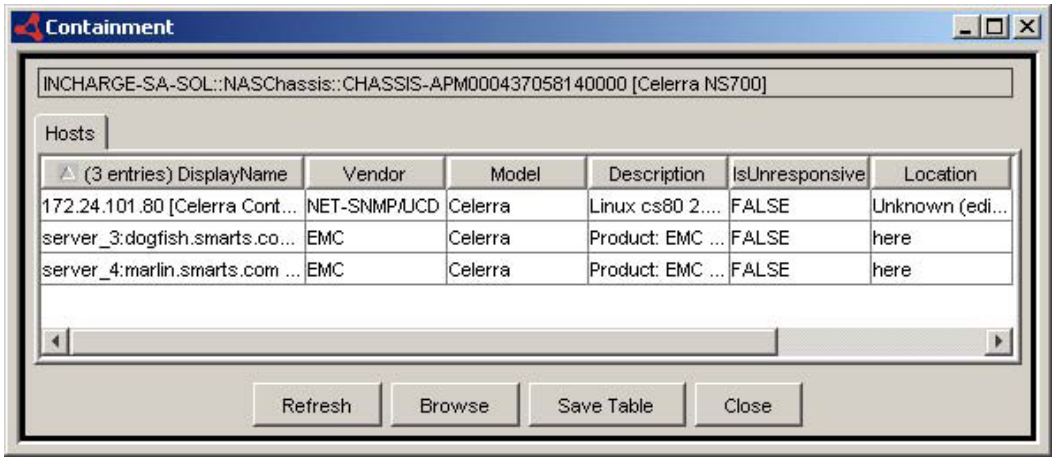
To obtain containment information about an individual NAS device element, you can use any of the following methods to open a Containment dialog for the element:

- In a NAS device map tree, right-click the element and select **Containment** in the pop-up menu.
- In a NAS device map display, right-click the element and select **Containment** in the pop-up menu.
- In a Topology Browser Console, right-click the element and select **Containment** in the pop-up menu.
- In a Notification Log Console, double-click a notification for the element and click the **Containment** button in the Notification Properties dialog.

NAS Containment dialog box

Figure 31 on page 174 shows a Containment dialog box for a NASChassis element, showing the associated network cards, adapters, IP addresses, and other information.

Figure 16-2. NASChassis Containment dialog box—example



Redundancy Group Creation

17

This chapter includes the following topics:

- Creation of card redundancy groups
- Creation of system redundancy groups
- Creation of network adapter redundancy groups

Creation of card redundancy groups

IP Availability Manager automatically creates CardRedundancyGroups during discovery for Cisco devices that support the CISCO-STACK-MIB. Each such group consists of two cards. You can view the card redundancy groups and their members by using the Domain Manager Administration Console.

In addition, you can create redundancy groups for cards that do not support the CISCO-STACK-MIB. The process for manually creating a card redundancy group is similar to that for creating a system redundancy group, as described in Creation of network adapter redundancy groups. You have to:

1 Create an instance of the CardRedundancyGroup class.

2 Insert the cards that participate in the redundancy group into the ComposedOf relationship.

Creation of system redundancy groups

Redundancy groups for network adapters, network connections, and systems must be manually created by using the **dmctl** utility or an Adapter Scripting Language (ASL) script.

To create a redundancy group for one of these types of objects:

1 Create an instance of one of the redundancy group classes.

2 Insert the objects that participate in the redundancy group into the ComposedOf relationship of the redundancy group.

The following example uses the **dmctl** utility to create an instance of a system redundancy group and inserts two members into the group:

```
/opt/InCharge7/IP/smarts/bin/dmctl -s INCHARGE-AM-PM
Server INCHARGE-AM-PM User: admin
admin's Password: XXXXXXXX
Domain Manager Control Program (V7.0.0.0) -- Type 'help' for a list of commands.
```

```
Attached to 'INCHARGE-AM-PM'
dmctl> create SystemRedundancyGroup::RouterRedundancyGroup
dmctl> insert SystemRedundancyGroup::RouterRedundancyGroup:: ComposedOf Router::router1.smarts.com
dmctl> insert SystemRedundancyGroup::RouterRedundancyGroup:: ComposedOf Router::router2.smarts.com
```

Network adapter, network connection, and system redundancy groups can be viewed through the Domain Manager Administration Console or the Topology Browser view of the Global Console.

Creation of network adapter redundancy groups

The status of a network adapter redundancy group changes when at least one adapter in the group is flapping or has a status of down, disabled, not present, or testing.

Note By default, a value of testing for the network adapter's status results in a Down notification. You can modify this behavior with the Testing Notification Mode.

In addition, you can create a network adapter redundancy group where the status calculation considers the state of the system containing the adapter. System status is useful when the group's members come from multiple systems. You determine the behavior by the topology you create. When all the network adapters in a group come from the same system, you can make the group PartOf a UnitaryComputerSystem. This prevents the unresponsive status of the system from affecting the status calculation.

The following examples use ASL code fragments to show several different network adapter redundancy groups.

Example 1

In example 1, both network adapters are part of the host named host_1. Because of this, the redundancy group is inserted into the PartOf relationship with the host. If host_1 is unresponsive, IP Availability Manager does not generate any notifications about the redundancy group. If the group is not part of the system, IP Availability Manager would notify AllComponentsDown when host_1 is unresponsive.

Also, note the use of the LayeredOver relationship from the network adapter redundancy group to the system. This information is used to draw maps of network adapter redundancy groups in the Global Manager:

```
rg = create("NetworkAdapterRedundancyGroup",
   My-Redundancy-Group");
//
// Turns off system status calc
rg->PartOf += object("Host", "host_1");
//
// For service map
rg->LayeredOver += object("Host", "host_1");
//
rg->ComposedOf += object("Interface", "IF-host_1/1");
rg->ComposedOf += object("Interface", "IF-host_1/2");
```

Example 2

In example 2, the network adapters are from different systems. Because of this, the redundancy group is not inserted into the PartOf relationship with any system. As a result, the status of this redundancy group is based on the status of the adapters and the responsive state of their containing systems:

```
rg = create("NetworkAdapterRedundancyGroup",
   My-Redundancy-Group");
//
rg->ComposedOf += object("Interface", "IF-host_1/1");
rg->ComposedOf += object("Interface", "IF-host_2/2");
//
// For service map
rg->LayeredOver += object("Host", "host_1");
rg-> LayeredOver += object("Host", "host_2");
```

In the next example, the network adapters are from a single system, but the system status is considered in the status calculation because the group is not a part of a system:

```
rg = create("NetworkAdapterRedundancyGroup",
   My-Redundancy-Group");
//
// For service map
rg->LayeredOver += object("Host", "host_3");
//
rg->ComposedOf += object("Interface", "IF-host_3/1");
rg->ComposedOf += object("Interface", "IF-host_3/2");
```

In the final example, the network adapters are from a single system but the system status is not included in the status calculation for the redundancy group. Again, this is because the group is inserted into the PartOf relationship with the system which consists of the network adapters.

```
rg = create("NetworkAdapterRedundancyGroup",
   My-Redundancy-Group");
//
// Turns off system status calc
rg->PartOf += object("Host", "host_4");
//
// For service map
rg->LayeredOver += object("Host", "host_4");
//
rg->ComposedOf += object("Interface", "IF-host_4/1");
rg->ComposedOf += object("Interface", "IF-host_4/2");
```