EMC² | smarts

MPLS Manager
1.2

CONFIGURATION GUIDE
P/N 300-002-532
REV A02

OL-8966-01

EMC Corporation
Corporate Headquarters:
Hopkinton, MA 01748-9103
1-508-435-1000
www.EMC.com

Contents

Preface

This document provides instructions for configuring the EMC Smarts MPLS Manager. Topics include editing control and configuration files, and configuring the EMC Smarts Service Assurance Manager (Global Manager) to work with the MPLS Manager.

Intended Audience

This document is intended for administrators and integrators who need to configure and maintain the MPLS Manager.

Prerequisites

Readers of this document should have a general understanding of UNIX and Windows systems. Readers must have root or Administrator privileges on the local system to perform the configurations.

Document Organization

This document consists of the following chapters and appendices.

Table 1: Document Organization

1. INTRODUCTION	Provides an architectural and functional overview of the MPLS Manager, and highlights configuration tasks for the MPLS Manager.
2. CONFIGURING THE MPLS MANAGER	Provides detailed information about configuring the MPLS Manager.
3. CONFIGURING THE GLOBAL MANAGER	Provides detailed information about configuring the Global Manager to work with the MPLS Manager.
A. UNDERSTANDING THE SM_EDIT UTILITY	Explains how to use the EMC Smarts *sm_edit* utility.
B. CONFIGURING A COMMUNITY STRING FOR JUNIPER PING MIB ACCESS	Explains how to configure a community string on Juniper devices so that the MPLS Manager can access their Ping MIBs.
C. WILDCARDS	Describes the wildcards used to create matching patterns.

Documentation Conventions

Several conventions may be used in this document as shown in Table 2.

Table 2: Documentation Conventions

CONVENTION	EXPLANATION
`sample code`	Indicates code fragments and examples in Courier font
keyword	Indicates commands, keywords, literals, and operators in bold
`%`	Indicates C shell prompt
#	Indicates C shell superuser prompt
<parameter>	Indicates a user-supplied value or a list of non-terminal items in angle brackets
[option]	Indicates optional terms in brackets

Table 2: Documentation Conventions *(continued)*

CONVENTION	EXPLANATION
/InCharge	Indicates directory path names in italics
yourDomain	Indicates a user-specific or user-supplied value in bold, italics
File > Open	Indicates a menu path in italics
▼▲	Indicates a command is wrapped over one or more lines. The command must be typed as one line.

Directory path names are shown with forward slashes (/). Users of the Windows operating systems should substitute back slashes (\) for forward slashes.

Also, if there are figures illustrating consoles in this document, they represent the consoles as they appear in Windows. Under UNIX, the consoles appear with slight differences. For example, in views that display items in a tree hierarchy such as the Topology Browser, a plus sign displays for Windows and an open circle displays for UNIX.

Finally, unless otherwise specified, the term InCharge Manager is used to refer to EMC Smarts programs such as Domain Managers, Global Managers, and adapters.

MPLS Management Suite Installation Directory

In this document, the term ***BASEDIR*** represents the location where EMC Smarts software is installed.

- For UNIX, this location is: */opt/InCharge<n>/<productsuite>*.
- For Windows, this location is: *C:\InCharge<n>\<productsuite>*.

The *<n>* represents the EMC Smarts software platform version number. The *<productsuite>* represents the InCharge product suite to which the product belongs. For example, on UNIX operating systems, MPLS Manager is installed to */opt/InCharge6/MPLS/smarts* by default. On Windows operating systems, this product is installed to *C:\InCharge6\MPLS\smarts* by default. This location is referred to as ***BASEDIR****/smarts*.

Optionally, you can specify the root of ***BASEDIR*** to be something other than */opt/InCharge6* (on UNIX) or *C:\InCharge6* (on Windows), but you cannot change the *<productsuite>* location under the root directory.

For more information about the directory structure of EMC Smarts software, refer to the *EMC Smarts System Administration Guide*.

MPLS Management Suite Products

The MPLS Management Suite offers the following products:

- MPLS Manager
- EMC Smarts Adapter for Cisco ISC
- Perl API

Additional Resources

In addition to this document, EMC Corporation provides the following resources.

Command Line Programs

Descriptions of command line programs are available as HTML pages. The *index.html* file, which provides an index to the various commands, is located in the ***BASEDIR****/smarts/doc/html/usage* directory.

Documentation

Readers of this document may find other documentation (also available in the ***BASEDIR****/smarts/doc/pdf* directory) helpful.

EMC Smarts Documentation

The following documents are product independent and thus relevant to users of all EMC Smarts products:

- *EMC Smarts Documentation Roadmap*
- *EMC Smarts System Administration Guide*
- *EMC Smarts ASL Reference Guide*
- *EMC Smarts Perl Reference Guide*

MPLS Management Suite Documentation

The following documents are relevant to users of the MPLS Management Suite product suite:

- *EMC Smarts MPLS Management Suite Installation Guide*
- *EMC Smarts MPLS Manager User's Guide*
- *EMC Smarts MPLS Manager Configuration Guide*
- *EMC Smarts MPLS Manager Discovery Guide Supplement*
- *EMC Smarts Adapter for Cisco ISC User's Guide*
- *EMC Smarts MPLS Management Suite Release Notes*

Refer to the *EMC Smarts Documentation Roadmap* for documentation resources provided with other EMC Smarts product suites.

Technical Support

For questions about technical support, call your local sales office or service provider. For service, call one of the following numbers:

United States: 800.782.4362 (SVC.4EMC)

Canada: 800.543.4782 (543.4SVC)

Worldwide: 508.497.7901

EMC Powerlink

EMC Powerlink is the EMC Corporation's secure extranet for customers and partners. Powerlink is an essential tool for obtaining web-based support from the EMC Corporation. Powerlink can be used to submit service or information requests (tickets) and monitor their progress, to review the knowledgebase for known problems and solutions, and to download patches and SmartPacks.

From training on EMC products and technologies, to online support, product announcements, software registration, technical white papers, interoperability information, and a range of configuration tools, Powerlink offers resources unavailable elsewhere.

For quickest access when you do not already have a Powerlink account, ask your EMC representative for the access code for your company and register at the Powerlink site. Visit the EMC Powerlink website at:

http://powerlink.emc.com

1

Introduction

The EMC Smarts MPLS Manager, working with EMC Smarts Availability Manager (Availability Manager) and EMC Smarts Service Assurance Manager (Global Manager), manages Multiprotocol Label Switching (MPLS) networks and Virtual Private Networks (VPNs) configured and provisioned over MPLS networks. In addition, the MPLS Manager works with optional, specialized adapters to synchronize provisioning data and events with provisioning systems, such as the Cisco IP Solution Center (ISC).

MPLS Architecture

Figure 1 illustrates the components of the MPLS architecture and the flow of information among the components:

- The Availability Manager performs discovery and polling of the underlying transport domain in the MPLS network, and sends topology and events to both the MPLS Manager and the Global Manager.
- The MPLS Manager performs discovery and polling of the MPLS network, and sends topology and events to the Global Manager.
- The Global Manager sends VPN data and notifications to the Global Console.

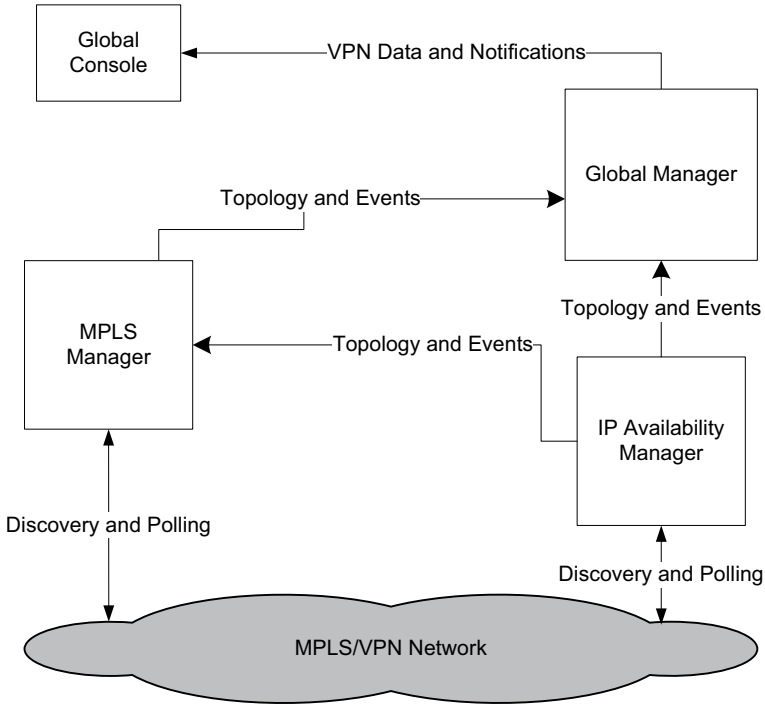

Figure 1: MPLS Architecture Topology and Events Flow

Figure 2 shows the same information flow with the addition of a customer provisioning system and customized adapter. In this case, the customer provisioning system sends VPN provisioning data and events to the customized adapter. The adapter communicates discovery and VPN provisioning data to and from the Global Manager, and the Global Manager sends the VPN provisioning data to the MPLS Manager and the Global Console.

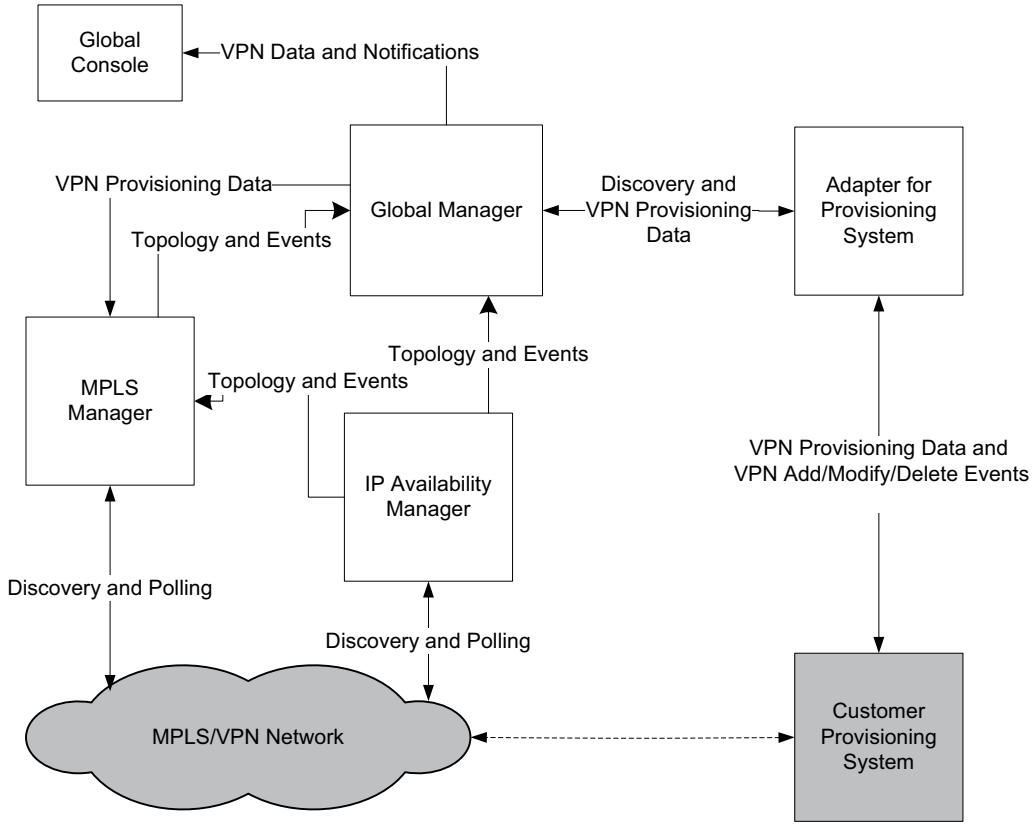

Figure 2: MPLS Architecture Supporting Customer Provisioning System

Availability Manager

The Availability Manager discovers physical and logical Layer 2 and Layer 3 network connectivity in multi-vendor, switched, and routed networks. It monitors and analyzes the network connectivity, sends network topology and event information to the Global Manager, and sends MPLS-relevant topology and event information to the MPLS Manager.

MPLS Manager

The MPLS Manager manages MPLS networks and MPLS Layer 3 and Layer 2 VPNs configured and provisioned over MPLS networks. It provides management capabilities for both the MPLS and the VPN domains of an MPLS network. It also provides for mapping and correlation among domains, and between the MPLS and VPN domains and the underlying transport domain, by means of cross-domain correlation and cross-domain impact analysis.

Global Manager

The Global Manager integrates the topology and event information imported from the Availability Manager and the MPLS Manager and relates the information to services and customers. It also provides cross-domain and end-to-end impact analysis.

The Global Manager displays the topology, event, and impact information through the Global Console.

Global Console

The Global Console enables users to browse the network protocol topology in various forms, including maps, and to view notifications about events that impact MPLS availability.

Configuration Roadmap

Configuring an MPLS Manager deployment involves configuring the MPLS Manager, the Global Manager, and the Availability Manager applications that are part of the deployment.

MPLS Manager Configuration Tasks

In addition to performing the configuration and administration tasks common to all Domain Managers, you perform the following additional (mostly optional) tasks to set up the MPLS Manager:

- Create Command Line Interface (CLI) login credentials (mandatory)
- Enable remote ping functionality
- Change remote ping global values
- Add Availability Managers as sources to the MPLS Manager
- Ensure that the Global Manager name is configured correctly
- Limit CLI discovery to specific routing devices or disable CLI discovery
- Preserve CLI log files across discovery cycles

All of these tasks are performed before the MPLS Manager is started. For detailed information about these configuration tasks, see *Configuring the MPLS Manager* on page 7.

Global Manager Configuration Tasks

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

- Edit the Service Assurance *ics.conf* file so that data from the Availability Manager and the MPLS Manager can be imported into the Global Manager.
- Add the Remote Ping Server Tools to the Global Manager and then to the appropriate User Profiles.

For detailed information about this configuration task, see *Configuring the Global Manager* on page 19.

Availability Manager Configuration Tasks

You perform no special tasks to configure and administer the Availability Manager in an MPLS Manager deployment. For general information about configuring the Availability Manager, see the *EMC Smarts IP Management Suite Deployment Guide*.

What's Next?

Upon configuring your MPLS Manager deployment, you are ready to begin the discovery. To understand, prepare for, and initiate MPLS Manager discovery, see the *EMC Smarts MPLS Manager Discovery Guide Supplement*.

2

Configuring the MPLS Manager

Before starting the MPLS Manager, edit configuration files to customize the MPLS Manager configuration for your environment using the procedures in this chapter.

Only one configuration procedure is required: you must create Command Line Interface (CLI) login credentials. The remaining configuration procedures are optional because the default configuration is appropriate for most deployments:

- Configure additional Availability Managers as sources of topology and events to the MPLS Manager (optional).
- Ensure that the Global Manager name is configured correctly (optional).
- Enable remote ping functionality (optional).
- Change remote ping global values (optional).
- Limit CLI discovery to specific routing devices or disable CLI discovery (optional).
- Preserve CLI log files across discovery cycles (optional).
- Customize L2VPN Functionality (optional).

All the procedures require that you modify one of the configuration files that are described in Table 3.

Use the *sm_edit* utility to edit these files. For information about *sm_edit*, see *Understanding the sm_edit Utility* on page 25.

Table 3: User-Editable Control and Configuration Files for the MPLS Manager

DIRECTORY UNDER BASEDIR	FILE NAME	DESCRIPTION
smarts/local/conf	*runcmd_env.sh*	Sets environment variables that enable remote ping functionality and specify CLI login credentials.
smarts/conf/mpls-vpn	*REMOTEPING.conf*	Configures global values assigned to newly deployed remote ping instances.
smarts/conf/mpls-vpn	*LOCAL.import*	Adds additional Availability Managers as sources, specifies Global Manager, limits or disables CLI discovery, preserves CLI log files across discovery cycles, configures L2VPN functionality.

Note: Before editing the files, ensure the appropriate patches are installed for the MPLS Manager. Check the *EMC Smarts MPLS Management Suite Installation Guide* for patch requirements.

Creating CLI Login Credentials (Required)

For the CLI discovery probe to establish Telnet sessions with the managed routing devices, you must add the CLI login environment variables identified in Table 4 to the *runcmd_env.sh* file. For MPLS CLI discovery to succeed, each managed routing device must be configured with the same set of credentials.

Table 4: CLI Login Environment Variables for CLI Discovery

ENVIRONMENT VARIABLE	DESCRIPTION
`EXPECT_USERID=<user_ID>`	User ID for the routing devices in the managed MPLS environment. A value is required. If the user ID is not configured on the devices, enter any value. For example, `EXPECT_USERID=xxx`.
`EXPECT_PASSWORD=<user_password>`	User password for the routing devices in the managed MPLS environment.
`EXPECT_ENABLE=<enable_password>` (Cisco only)	The enable password that provides administrative access to the Cisco routing devices in the managed MPLS environment. A value is required. If the enable password is not configured on the Cisco devices, enter any value. For example, `EXPECT_ENABLE=xxx`.

To create CLI login credentials, follow these steps:

1. Go to the ***BASEDIR***/*smarts/bin* directory in the MPLS Manager installation area and open *runcmd_env.sh* using the *sm_edit* utility:

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

2. Add the following lines to the file:

   ```
   EXPECT_USERID=<user ID>
   EXPECT_PASSWORD=<user password>
   EXPECT_ENABLE=<enable password>
   export EXPECT_USERID EXPECT_PASSWORD EXPECT_ENABLE
   ```

   Replace <***user ID***>, <***user password***>, and <***enable password***> with the appropriate values for your deployment, as described in Table 4. For example:

   ```
   EXPECT_USERID=dorado
   EXPECT_PASSWORD=dorado
   EXPECT_ENABLE=*
   export EXPECT_USERID EXPECT_PASSWORD EXPECT_ENABLE
   ```

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

The *runcmd_env.sh* file automatically sets the CLI login environment variables (and any other environment variable definitions that the file contains) for each newly started MPLS Manager application.

For information about CLI discovery, see the *EMC Smarts MPLS Manager Discovery Guide Supplement*.

Configuring Additional Availability Managers as Sources of Topology and Events (Optional)

By default, the MPLS Manager is configured to add an Availability Manager named INCHARGE-AM as a source for topology and events. The *LOCAL.import* file can be edited to change this name and/or configure one or more additional Availability Managers as sources. The block of lines to be edited is:

```
InChargeDomain::InChargeDomain_INCHARGE-AM {
       Type = "AM"
       DomainName = "INCHARGE-AM"
       DisplayName = "INCHARGE-AM"
}
```

To change the default Availability Manager name or to configure an additional Availability Manager as a source of topology and events to the MPLS Manager, follow these steps:

1. Go to the ***BASEDIR****/smarts/bin* directory in the MPLS Manager installation area and open *LOCAL.import* using the *sm_edit* utility:

   ```
   # ./sm_edit conf/mpls-vpn/LOCAL.import
   ```

2. Find the section, *Instances of AM Server*, and go to the section below the comments and find the following five lines:

   ```
   InChargeDomain::InChargeDomain_INCHARGE-AM {
          Type = "AM"
          DomainName = "INCHARGE-AM"
          DisplayName = "INCHARGE-AM"
   }
   ```

3. On the `InChargeDomain` and `DomainName` lines, change `INCHARGE-AM` to the name of your Availability Manager. On the `DisplayName` line, change `INCHARGE-AM` to the name that you want listed for the Availability Manager in the topology. For example:

   ```
   InChargeDomain::InChargeDomain_ASIA-AM1 {
          Type = "AM"
          DomainName = "ASIA-AM1"
          DisplayName = "ASIA-AM1"
   }
   ```

4. To configure additional Availability Manager sources, copy and paste additional versions of the `InChargeDomain` section (five lines) and then follow the directions given in Step 3.

5. Save the *LOCAL.import* file.

6. If the MPLS Manager was running before you edited the *LOCAL.import* file, restart the MPLS Manager.

At startup, the MPLS Manager reads the *LOCAL.import* file, saves the configuration information in its repository, and imports all routers discovered by the Availability Manager sources.

Ensuring that the Global Manager Name is Configured Correctly (Optional)

By default, the MPLS Manager is configured to communicate with a Global Manager named INCHARGE-SA. If your Global Manager is named differently, you must change the value in the *LOCAL.import* file:

1. Go to the ***BASEDIR****/smarts/bin* directory in the MPLS Manager installation area and open *LOCAL.import* using the *sm_edit* utility:

   ```
   # ./sm_edit conf/mpls-vpn/LOCAL.import
   ```

2. Find the section, *Instance of SAM Server*, and change the Global Manager name:

   ```
   ICSDomain::<GLOBAL_MANAGER-NAME>
   ```

   Where ***<GLOBAL_MANAGER-NAME>*** is the name used for the Global Manager in your deployment.

3. Save the *LOCAL.import* file.

4. If the MPLS Manager was running before you edited the *LOCAL.import* file, restart the MPLS Manager.

At startup, the MPLS Manager reads the *LOCAL.import* file.

Enabling Remote Ping Functionality (Optional)

By default, remote ping functionality is disabled for the MPLS Manager. Enabling remote ping functionality requires adding the following environment variable to the *runcmd_env.sh* file:

SM_ENABLE_SNMP_SET=TRUE

To enable remote ping functionality, follow these steps:

1. Go to the ***BASEDIR****/smarts/bin* directory in the MPLS Manager installation area and open *runcmd_env.sh* using the *sm_edit* utility:

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

2. Add the following lines to the file:

   ```
   SM_ENABLE_SNMP_SET=TRUE
   export SM_ENABLE
   ```

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

The *runcmd_env.sh* file automatically sets the ENABLE_SNMP_SET environment variable (and any other environment variables that the file contains) for each newly started MPLS Manager application.

For information about deploying a periodic remote ping instance or invoking an on-demand remote ping, see the *EMC Smarts MPLS Manager User's Guide*.

Changing Remote Ping Global Values (Optional)

Except where noted in Table 5, each newly deployed periodic remote ping instance or newly invoked on-demand remote ping inherits the global values specified in the *REMOTEPING.conf* file.

Table 5: Global Values in the *REMOTEPING.conf* File

PARAMETER	TYPE	DEFAULT GLOBAL VALUE	ALLOWED VALUES	DESCRIPTION
MibStartIndex	Integer	1	Integer greater than or equal to 1	In the PING MIB table, the starting location of rows reserved for the MPLS Manager's remote ping requests.
MibBlockSize	Integer	1000	1 to 1000	In the PING MIB table, the number of rows reserved for the MPLS Manager's remote ping requests.
WriteCommunity*	String	"private"	String of unspecified length	Community name used by the MPLS Manager to write (SET) ping requests to the PING MIB table.
PingsPerInterval	Integer	4	4 to 10	Number of pings sent by the MPLS Manager per polling interval.
ImpairedThreshold**	Integer	2	Integer greater than or equal to 1	Minimum number of missed packets before the MPLS Manager generates an *Impaired* notification.

* Typically overwritten by a value defined in a *System Write Community Strings* group.

** Not applicable to on-demand remote ping.

The WriteCommunity value defined in the *REMOTEPING.conf* file is included in a remote ping request *only* if the MPLS Manager cannot find a match for the target router in a *System Write Community Strings* group. For information about modifying or creating a *System Write Community Strings* group, see the *EMC Smarts MPLS Manager User's Guide*. For information about configuring a community string for Juniper Ping MIB access, see *Configuring a Community String for Juniper Ping MIB Access* on page 27.

To change the remote ping global values, follow these steps:

1. Go to the ***BASEDIR****/smarts/bin* directory in the MPLS Manager installation area and open *REMOTEPING.conf* using the *sm_edit* utility:

   ```
   # ./sm_edit conf/mpls-vpn/REMOTEPING.conf
   ```

2. Make your changes to the global values, as described in Table 5.

3. Save the *REMOTEPING.conf* file. The modified version of the file is saved to the ***BASEDIR****/smarts/local/conf/mpls-vpn* directory.

Upon saving your edits, any newly deployed periodic remote ping instance or newly invoked on-demand remote ping will inherit your changed global values.

Limiting or Disabling CLI Discovery (Optional)

The MPLS Manager is configured with default settings that support discovery of all routing devices in the managed network. To limit CLI discovery to specific routing devices or, to disable CLI discovery, set parameters as described in the *LOCAL.import* file, Table 6.

Table 6: CLI Discovery Type Parameters in the *LOCAL.import* File

PARAMETER	VALUES	DESCRIPTION
CLIProhibit	TRUE, FALSE Default: FALSE	Determines whether CLI discovery is disabled. If `CLIProhibit = FALSE`, CLI discovery is enabled. If `CLIProhibit = TRUE`, CLI discovery is disabled.
CLIFilter	Regular expression Default: `"*"`	Specifies the wildcard expression to use to limit CLI discovery to specific devices. Use wildcards to specify ranges of IP addresses or ranges of names. See *Wildcards* on page 29 for more information.
CLIFilterType	"CLI_AGENTADDRESS", "CLI_SYSTEMNAME" Default: "CLI_AGENTADDRESS"	Specifies the type of filter to use for the CLI filter. The value depends on the value of the `CLIFilter` parameter. If IP addresses are specified, set `CLIFilterType = "CLI_AGENTADDRESS".` If names are specified, set `CLIFilterType = "CLI_SYSTEMNAME".`

Limiting CLI Discovery to Specific Routing Devices

1. Go to the ***BASEDIR****/smarts/bin* directory in the MPLS Manager installation area and open *LOCAL.import* using the *sm_edit* utility:

   ```
   # ./sm_edit conf/mpls-vpn/LOCAL.import
   ```

2. At the top of the *LOCAL.import* file, uncomment the following lines (shown uncommented) for the `MPLS-MANAGER` definition:

   ```
   CLIFilter = "*"
   CLIFilterType "CLI_AGENTADDRESS"
   ```

Note: CLI discovery is enabled by default.

3 For the `CLIFilter` parameter, replace the asterisk with the device IP address(es) or name(s) to be discovered. Use wildcards to specify ranges of IP addresses or ranges of names; for example, `172.16.*` or `R1*`.

   For `172.16.*`, only routing devices having loopback IP addresses beginning with the string `172.16` will be members of the group; for `R1*`, only routing devices having names beginning with the string `R1` will be members of the group. For information about EMC Smarts wildcard patterns, see *Wildcards* on page 29.

4 Based on whether IP addresses or names were specified in the previous step, set the `CLIFilterType` parameter to the appropriate option, as indicated below:

   - IP addresses–Use `CLI_AGENTADDRESS`
   - Names–Use `CLI_SYSTEMNAME`

5 Save the *LOCAL.import* file. The modified version of the file is saved to the ***BASEDIR****/smarts/local/conf/mpls-vpn* directory.

6 If the MPLS Manager was running before you edited the *LOCAL.import* file, restart the MPLS Manager.

Disabling CLI Discovery

1 Go to the ***BASEDIR****/smarts/bin* directory in the MPLS Manager installation area and open *LOCAL.import* using the *sm_edit* utility:

```
# ./sm_edit conf/mpls-vpn/LOCAL.import
```

2 At the top of the *LOCAL.import* file, uncomment the following line (shown uncommented) for the `MPLS-MANAGER` definition:

```
CLIProhibit = TRUE
```

Note: `CLIProhibit = TRUE` disables CLI discovery.

3 Save the *LOCAL.import* file. The modified version of the file is saved to the ***BASEDIR****/smarts/local/conf/mpls-vpn* directory.

4 If the MPLS Manager was running before you edited the *LOCAL.import* file, restart the MPLS Manager.

At startup, the MPLS Manager reads the *LOCAL.import* file and saves the configuration information in its repository.

Preserving CLI Log Files Across Discovery Cycles (Optional)

By default, the MPLS Manager is configured to delete all CLI log files in the ***BASEDIR****/smarts/local/logs* directory just before performing a discovery cycle. To change this behavior and preserve CLI log files across discovery cycles, follow these steps:

1. Go to the ***BASEDIR****/smarts/bin* directory in the MPLS Manager installation area and open *LOCAL.import* using the *sm_edit* utility:

   ```
   # ./sm_edit conf/mpls-vpn/LOCAL.import
   ```

2. Find the `RemoveExpectLogs` parameter:

   ```
   RemoveExpectLogs = TRUE
   ```

3. Change the value of the `RemoveExpectLogs` parameter to `FALSE`.

4. Save the *LOCAL.import* file. The modified version of the file is saved to the ***BASEDIR****/smarts/local/conf/mpls-vpn* directory.

5. If the MPLS Manager was running before you edited the *LOCAL.import* file, restart the MPLS Manager.

Customizing L2VPN Functionality

The *LOCAL.import* file includes parameters for customizing L2VPN operational. Use the following procedure to disable, restrict, or modify the behavior of the default L2VPN functionality.

1. Go to the ***BASEDIR****/smarts/bin* directory in the MPLS Manager installation area and open *LOCAL.import* using the *sm_edit* utility:

   ```
   # ./sm_edit conf/mpls-vpn/LOCAL.import
   ```

2. Find the desired L2VPN parameter and enter the appropriate value as described in Table 7. If the parameter line is also commented out, delete the # character.

Table 7: L2VPN Parameters in the LOCAL.import File

PARAMETER	VALUES	DESCRIPTION
L2VPN_CLI_EXISTS	TRUE, FALSE Default: TRUE	Determines if CLI-based probe for L2VPNs is attempted on Cisco devices. If you are sure that the Cisco devices in your managed network do not support L2VPN, set to FALSE.
L2VPN_ENABLED	TRUE, FALSE Default: FALSE	Determines if L2VPN feature is enabled on the MPLS Manager. by default, the feature is disabled. To enable the functionality, set to TRUE.
L2VPN_CREATE_VPN	TRUE, FALSE Default: TRUE	Determines if a VPN is created for each discovered Martini-implemented L2VPN. By default, they are created. For a VPLS environment, the recommended setting is FALSE.

3. Save the *LOCAL.import* file. The modified version of the file is saved to the ***BASEDIR****/smarts/local/conf/mpls-vpn* directory.
4. If the MPLS Manager was running before you edited the *LOCAL.import* file, restart the MPLS Manager.

Security

The security configuration files, *clientConnect.conf* and *serverConnect.conf*, allow you to set up secure connections between the components in an MPLS Manager deployment. By default, the configuration option settings in the *clientConnect.conf* file and the *serverConnect.conf* files allow minimally secure connections between the components.

For detailed information about SMARTS secure communications, see the *EMC Smarts System Administration Guide*.

3

Configuring the Global Manager

Configuring the Global Manager for MPLS Manager requires these tasks:

- For the Global Manager to import MPLS Manager topology and events, you must configure an MPLS Manager DomainType definition in the Global Manager *ics.conf* file. In addition, if you have not already done so, you must configure a DomainType definition for each Availability Manager that will be part of the deployment. The *ics.conf* file is located in the ***BASEDIR****/smarts/conf/ics* directory of the Global Manager installation area.
- To provide Global Console users with access to the Remote Ping server tools, ensure that the Remote Ping server tools are installed at the Global Manager. Then assign the tools to the appropriate User Profiles.

Editing the *ics.conf* File

Before editing the *ics.conf* file, ensure that the appropriate version of the Global Manager is installed with the appropriate SmartPack and/or rolling patch. Check the *EMC Smarts MPLS Management Suite Installation Guide* for version and patch requirements.

To edit the *ics.conf* file, follow these steps:

1. Go to the ***BASEDIR****/smarts/bin* directory in the Global Manager installation area and open *ics.conf* using the *sm_edit* utility:

   ```
   # ./sm_edit conf/ics/ics.conf
   ```

2. Find the MPLS Manager DomainType definition in the file and make the following changes:

   - Uncomment the MPLS Manager DomainType definition, or add an uncommented MPLS Manager DomainType definition to your file. (Do not uncomment the HookScript field unless you customize the associated ASL hookscript file.)
   - Change the name in the Name field to the name of your MPLS Manager. The default name is `INCHARGE-MPLS`.

   The following is an example of the DomainType definition for an MPLS Manager:

   ```
   #   DomainType definition for MPLS.
    DomainType
    {
        ConfFile          = "dxa-mpls-vpn-ics.conf";
        MinimumCertainty  = 0.24;
        SmoothingInterval = 65;
   ##   HookScript        = "ics/dxa-sample-hook.asl";
        Name              = "INCHARGE-MPLS";
    }
   ```

3. Find the Availability Manager DomainType definition and, if necessary, make the following changes:

   - Uncomment the Availability Manager DomainType definition, or add an uncommented Availability Manager DomainType definition to your file. (Do not uncomment the HookScript field unless you customize the associated ASL hookscript file.)
   - Change the name in the Name field to the name of your Availability Manager. The default name is `INCHARGE-AM`.

The following is an example of the DomainType definition for an Availability Manager:

```
#   DomainType definition for INCHARGE-AM.
 DomainType
 {
    ConfFile          = "dxa-conn.conf";
    MinimumCertainty  = 0.24;
    SmoothingInterval = 65;
#        HookScript         = "ics/dxa-sample-hook.asl";
    Name              = "INCHARGE-AM";
 }
```

4. If your deployment contains multiple Availability Managers, copy and paste multiple versions of the Availability Manager DomainType definition—one section per Availability Manager—and make the changes described in Step 3.
5. Save the *ics.conf* file. The modified version of the file is saved to the ***BASEDIR****/smarts/local/conf/ics* directory.

Forcing the Global Manager to Read the *ics.conf* File

To force the Global Manager to read the edited *ics.conf* file, follow these steps:

1. Go to the ***BASEDIR****/smarts/bin* directory in the Global Manager installation area.
2. Enter the following command:

```
▼# ./sm_adapter -s <GLOBAL_MANAGER-NAME>
      ../rules/ics/ICS_RemoteConfig.asl▲
```

▼▲ Indicates that the command must be typed as one line.

You will prompted for a user name and password for the Global Manager. For more information about the *ics.conf* file, see the *EMC Smarts System Administration Guide* and the *EMC Smarts Service Assurance Manager Configuration Guide*.

Installing and Configuring Remote Ping Server Tools

To provide Global Console users with access to the Remote Ping server tools, ensure that the Remote Ping server tools are installed at the Global Manager. Then assigned the tools to the appropriate User Profiles.

1. Go to the ***BASEDIR****/smarts/bin* directory in the Global Manager installation area.
2. Enter the following command:

   ▼`# ./sm_config --server=<GLOBAL_MANAGER-NAME> import --force mpls-action-config.xml`▲
3. Start the Global Console, or, if it is already running, restart it. The Topology Browser Console opens.
4. In the Topology Browser Console, select *Configure > Global Manager Administration Console*. This requires an EMC Smarts user account with the following privileges and permissions:
   - *All* privileges, specified in the *serverConnect.conf* file (or its equivalent) read by the Global Manager.
   - Permission to use the console operation *Configure Global Manager Admin Console*. Through the Global Manager Administration Console, this permission is specified in the Console Operations section of the user profile.

   For information about configuring access privileges and permissions to perform console operations, see the *EMC Smarts System Administration Guide* and *EMC Smarts Service Assurance Manager Configuration Guide*, respectively.
5. In the Global Manager Administration Console, add the Remote Ping server tools to the administration user profile:
   - Select **User Profiles** in the tree.
   - Select **admin-profile**.
   - From the Configure User Profile panel, click **Modify List** in the Server Tools section. This displays the Modify Server Tools dialog box.
   - Select all the Remote Ping server tools and add them to the profile by clicking **Add**.
   - Click **OK** to close the dialog box and click **Apply** at the bottom of the Configure User Profile panel.

Note: The Remote Ping server tools can also be added to other user profiles. As Remote Ping server tools use network resources, the tools should only be made accessible to users who understand the implications of their use.

A

Understanding the sm_edit Utility

As part of the EMC Smarts deployment and configuration process, you will need to modify certain files. User modifiable files include configuration files, rule set files, templates, and files (such as seed files, and security configuration files) containing encrypted passwords. Original versions of these files are installed into appropriate subdirectories under the ***BASEDIR****/smarts/* hierarchy. For example, on UNIX operating systems the original versions of Global Manager configuration files are installed to */opt/InCharge6/SAM/smarts/conf/ics*.

Original versions of files should not be altered. If a file requires modification, it must be stored as a local copy of the file in ***BASEDIR****/smarts/local* or one of its subdirectories. For example, a modified *ics.conf* file should be saved to */opt/InCharge6/SAM/smarts/local/conf/ics*. EMC Smarts software is designed to first search for user modifiable files in ***BASEDIR****/smarts/local* or one of its subdirectories. If a modified version of a file is not found in the local area, EMC Smarts software then searches appropriate nonlocal directories.

Note: Original versions of files may be changed or updated as part of an EMC Smarts software upgrade. However, files located in ***BASEDIR****/smarts/local* are always retained during an upgrade.

To facilitate proper file editing, EMC Corporation provides the *sm_edit* utility with every EMC Smarts product suite. When used to modify an original version of a file, this utility automatically creates a local copy of the file and places it in the appropriate location under ***BASEDIR****/smarts/local*. This ensures that the original version of the file remains unchanged. In both UNIX and Windows environments, you can invoke *sm_edit* from the command line. Optionally, you can configure Windows so that *sm_edit* is automatically invoked when user-modifiable files are double-clicked in Windows Explorer.

To invoke the *sm_edit* utility from the command line, specify the path and the name of the file you want to edit under ***BASEDIR****/smarts*. If multiple EMC Smarts products are running on the same host, you should ensure that you invoke *sm_edit* from the *bin* directory of the product suite whose files you wish to edit. For example, to edit the configuration file for the Global Manager, you invoke the *sm_edit* utility as follows:

```
# /opt/InCharge6/SAM/smarts/bin/sm_edit conf/ics/ics.conf
```

The *sm_edit* utility automatically creates a local copy of the *ics.conf* file in the ***BASEDIR****/smarts/local/conf/ics* directory, if necessary, and opens the file in a text editor. If a local version of the file already exists, the *sm_edit* utility opens the local version in a text editor. In addition, *sm_edit* creates any necessary directories.

For more information about how to properly edit user modifiable files and how to use the *sm_edit* utility, refer to the *EMC Smarts System Administration Guide*.

B

Configuring a Community String for Juniper Ping MIB Access

The default SNMP community strings for Juniper routing devices do not allow remote ping operations, so you must create an additional community string on the managed Juniper devices so that the MPLS Manager can access their Ping MIBs.

To define the additional community string, use the Juniper Command Line Interface (CLI) to access the Juniper devices and to specify the following candidate configuration:

```
snmp {
    view <view-name> {
        oid <ping-mib-object-identifier> include;
    }
    community <community-name> {
        authorization read-write;
        view <view-name>;
    }
} # End of [edit snmp] hierarchy level
```

After creating the candidate configuration, commit the configuration to the JUNOS Internet software running on the Juniper devices.

For example, the following candidate configuration creates a community string named `remote-ping-community` that grants all SNMP clients, such as the MPLS Manager, read-write access to the `DISMAN-PING-MIB` and the `JUNIPER-PING-MIB`:

```
snmp {
    view remote-ping-view {
        oid 1.3.6.2.1.80 include;       # DISMAN-PING-MIB
        oid 1.3.6.1.4.2636.3.7 include; # JUNIPER-PING-MIB
    }
    community remote-ping-community {
        authorization read-write;
        view remote-ping-view;
    }
} # End of [edit snmp] hierarchy level
```

Once the example configuration is committed to a Juniper device, the device will respond to SNMP Get, GetNext, GetBulk, and Set requests that contain the community string `remote-ping-community` and specify an object identifier (OID) having a `1.3.6.2.1.80` or `1.3.6.1.4.2636.3.7` prefix.

For more information about configuring Juniper devices, see the *JUNOS Network Management Configuration Guide*.

C

Wildcards

A wildcard pattern is a series of characters that are matched against incoming character strings. You can use these patterns when you define pattern matching criteria.

Matching is done strictly from left to right, one character or basic wildcard pattern at a time. Basic wildcard patterns are defined in Table 8. Characters that are not part of match constructs match themselves. The pattern and the incoming string must match completely. For example, the pattern *abcd* does not match the input *abcde* or *abc*.

A compound wildcard pattern consists of one or more basic wildcard patterns separated by ampersand (&) or tilde (~) characters. A compound wildcard pattern is matched by attempting to match each of its component basic wildcard patterns against the entire input string. For compound wildcard patterns, see Table 9.

If the first character of a compound wildcard pattern is an ampersand (&) or tilde (~) character, the compound is interpreted as if an asterisk (*) appeared at the beginning of the pattern. For example, the pattern ~*[0-9]* matches any string not containing any digits. A trailing instance of an ampersand character (&) can only match the empty string. A trailing instance of a tilde character (~) can be read as “except for the empty string.”

Note: Spaces are interpreted as characters and are subject to matching even if they are adjacent to operators like "&".

Table 8: Basic Wildcard Patterns

CHARACTER	DESCRIPTION
Note: Spaces specified before or after wildcard operators are interpreted as characters and are subject to matching.	
?	Matches any single character. For example, *server?.smarts.com* matches *server3.smarts.com* and *serverB.smarts.com*, but not *server10.smarts.com*.
*	Matches an arbitrary string of characters. The string can be empty. For example, *server*.smarts.com* matches *server-ny.smarts.com* and *server.smarts.com* (an empty match).
[set]	Matches any single character that appears within [set]; or, if the first character of [set] is (^), any single character that is *not* in the set. A hyphen (-) within [set] indicates a range, so that [*a-d*] is equivalent to [*abcd*]. The character before the hyphen (-) must precede the character after it or the range will be empty. The character (^) in any position except the first, or a hyphen (-) at the first or last position, has no special meaning. Example, *server[789-].smarts.com* matches *server7.smarts.com* through *server9.smarts.com*, but not *server6.smarts.com*. It also matches *server-.smarts.com*. Example: *server[^12].smarts.com* does not match *server1.smarts.com* or *server2.smarts.com*, but will match *server8.smarts.com*.
<n1-n2>	Matches numbers in a given range. Both *n1* and *n2* must be strings of digits, which represent non-negative integer values. The matching characters are a non-empty string of digits whose value, as a non-negative integer, is greater than or equal to *n1* and less than or equal to *n2*. If either end of the range is omitted, no limitation is placed on the accepted number. For example, *98.49.<1-100>.10* matches a range of IP addresses from *98.49.1.10* through *98.49.100.10*. Example of an omitted high end of the range: *<50->* matches any string of digits with a value greater than or equal to 50. Example of an omitted low end of the range: *<-150>* matches any value between zero and 150. A more subtle example: The pattern *<1-10>** matches 1, 2, up through 10, with * matching no characters. Similarly, it matches strings like 9x, with * matching the trailing x. However, it does not match 11, because <1-10> always extracts the longest possible string of digits (11) and then matches only if the number it represents is in range.

Table 8: Basic Wildcard Patterns *(continued)*

CHARACTER	DESCRIPTION
|	Matches alternatives. For example, "*ab|bc|cd*" without spaces matches exactly the three following strings: "*ab*", "*bc*", and "*cd*". A | as the first or last character of a pattern accepts an empty string as a match. Example with spaces "*ab | bc*" matches the strings "*ab*" and "*bc*".
\	Removes the special status, if any, of the following character. Backslash (\) has no special meaning within a set ([set]) or range (<n1-n2>) construct.

Special characters for compound wildcard patterns are summarized below.

Table 9: Compound Wildcard Patterns

CHARACTER	DESCRIPTION
&	"And Also" for a compound wildcard pattern. If a component basic wildcard pattern is preceded by & (or is the first basic wildcard pattern in the compound wildcard pattern), it *must* successfully match. Example: *NY*&*Router* matches all strings which contain NY and also contain Router. Example: <1-100>&*[02468] matches even numbers between 1 and 100 inclusive. The <1-100> component only passes numbers in the correct range and the *[02468] component only passes numbers that end in an even digit. Example: *A*|*B*&*C* matches strings that contain either an A or a B, and also contain a C.
~	"Except" for a compound wildcard pattern (opposite function of &).If a component basic wildcard pattern is preceded by ~, it *must not* match. Example: 10.20.30.*~10.20.30.50 matches all devices on network 10.20.30 except 10.20.30.50. Example: *Router*~*Cisco*&*10.20.30.*~10.20.30.<10-20>* matches a Router, except a Cisco router, with an address on network 10.20.30, except not 10.20.30.10 through 10.20.30.20.

Index

S

T

V

W