CA Spectrum®

Secure Domain Manager User Guide

Release 9.3

CA Technologies Product References

This document references the following CA Technologies products:

- CA Spectrum® (CA Spectrum)
- CA Spectrum® Secure Domain Manager
- CA Spectrum® Secure Domain Connector

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: Introduction

This section contains the following topics:

Challenges to Managing Highly Secure Networks

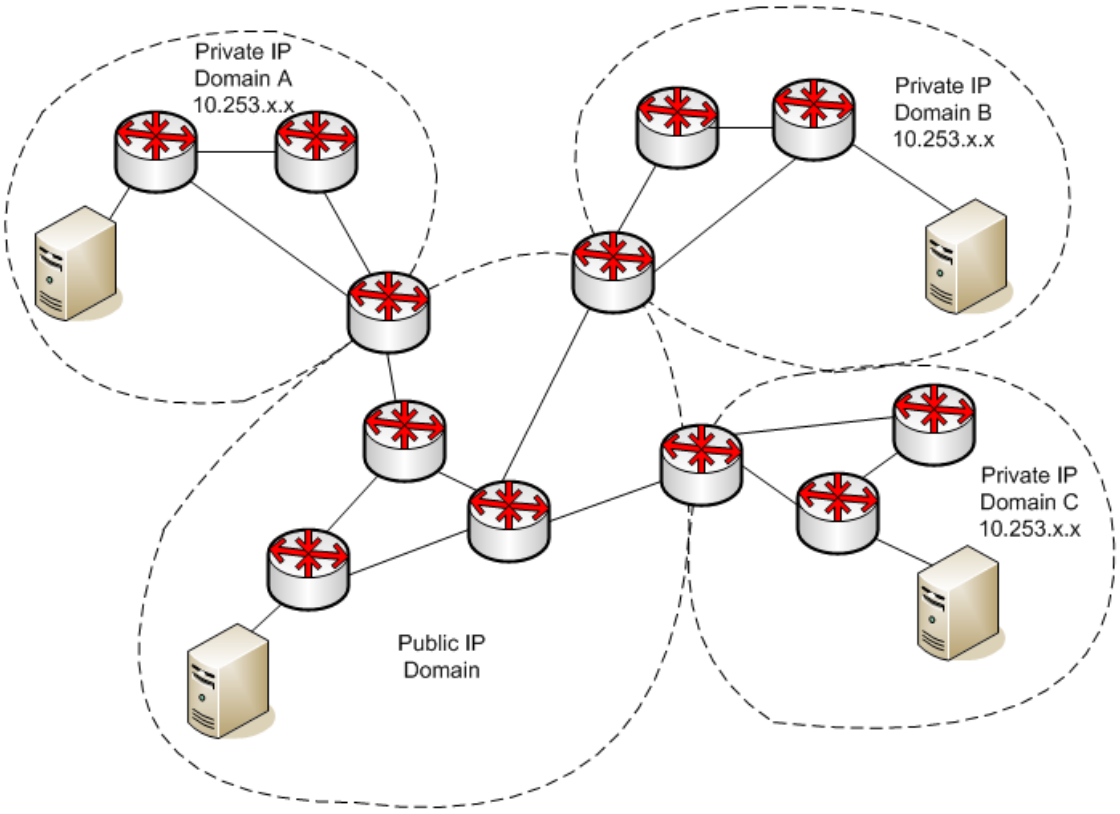
The computer networks are more secure these days. The challenges that are involved in managing devices and applications in secure networks are correspondingly greater. These challenges include:

- Managing network elements in overlapping (or private) IP domains (NAT environments)
- Managing network elements behind firewalls that is configured to block SNMP and ICMP traffic
- Managing network elements across insecure network domains

The Secure Domain Manager product provides a unique solution to these management challenges.

Overlapping IP Domains

The following diagram shows a NAT network that contains a public IP domain and three private IP domains that contain the same IP subnets.



Domains can represent the managed network of a company, a newly acquired division of a large enterprise, or a managed wireless hot spot in an airport terminal.

The following types of CA Spectrum customers face the overlapping IP challenge:

Managed service providers (MSPs)

MSPs use CA Spectrum to manage the networks of other organizations. The customers that MSPs manage invariably use IP ranges typically used for private IPs, 10.x.x.x or 172.16.x.x, for example. Therefore, the MSP must address the challenge of managing duplicate or overlapping IP addresses. In the past, this challenge was addressed by dedicating a CA Spectrum management server (the SpectroSERVER) to each customer that was using the same IP address space.

This posed two issues. The first involved cost. A dedicated management server was necessary for each customer, regardless of the size of the managed environment and the number of overlapping IP addresses it used. The second issue involved administration. The MSP was burdened with maintaining more management systems. MSPs required less expensive and efficient alternatives to dedicated management systems, especially when the number of elements with overlapping IP addresses was small and did not warrant the expense of a dedicated management server.

Hotspot (Wi-Fi) access providers

Hotspot access providers provide Wi-Fi access in locations such as airport terminals, airport lounges, hotel rooms, and coffee shops. For each location, the same private IP address space is issued. This approach simplifies configuration, installation, and administration. A provider may have hundreds or thousands of hotspots. To deploy a new hotspot quickly, each set of equipment that establishes the hotspot in a property is configured identically, including the IP address space. Once the hotspot is up and running, the challenge becomes managing it proactively to sustain an optimal level of service.

Enterprise managers

In an organization merger or acquisition scenario, an enterprise management staff must typically combine two entirely different and separately constructed IP networks, in many instances resulting in multiple overlapping IP addresses. In this scenario, the new IT organization must now deal with managing the combined network, especially the management of a network with the same IP address spaces. One solution to this challenge is to reassign IPs to every IP entity so there is no duplication of IP addresses. That solution involves a huge undertaking that presents many challenges.

Secure Domain Manager lets these customers overcome the challenge of managing overlapping IP domains in the following ways:

- Allows MSPs to deploy only a single, lightweight agent process on a host machine in each customer's remote network, thus removing the need to deploy and administer a full CA Spectrum installation.
- Allows the Hotspot access providers and large enterprises keep the overlapping private IP domains intact and manage the networks by using a lightweight agent process.

Firewalls Blocking SNMP and ICMP Traffic

Firewalls provide the security crucial to many network environments. Some challenges exist in managing a network behind a firewall. First, network administrators often configure firewalls to block SNMP and ICMP traffic, which provide the visibility into their network infrastructure, to unauthorized sources. Second, the configuration that is required to manage network elements through a highly secure firewall is complicated. Because, all the hosts and ports that are involved are identified and opened on the firewall to enable full management capabilities.

Secure Domain Manager allows network managers to overcome the challenge of managing networks through secure firewalls in the following ways:

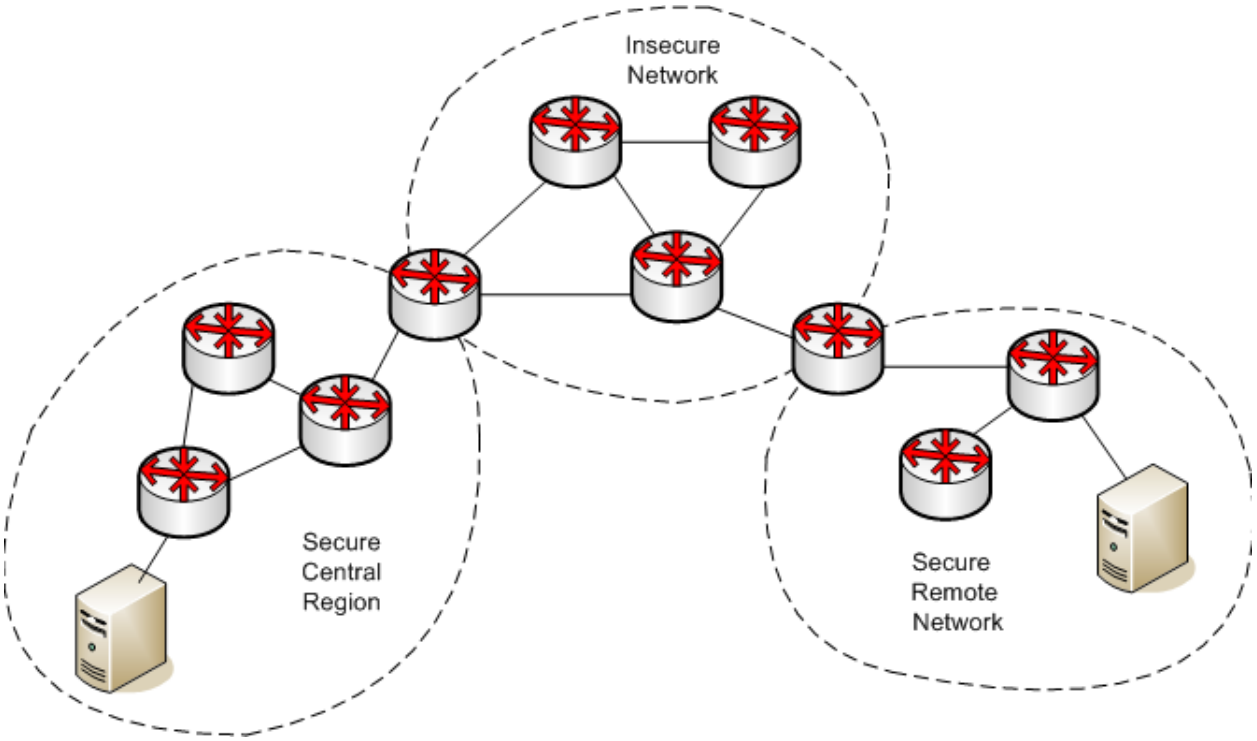
- Encoding UDP-based SNMP and the ICMP packets into a TCP/IP based protocol to overcome the restriction of the firewall on SNMP and ICMP traffic.
- Simplifying the configuration of the firewall by opening a single port that allows SNMP and ICMP traffic to flow between two well-defined hosts, on a well-defined port.

SNMP Traffic Passing Across Insecure Networks

SNMPv1 and SNMPv2 are insecure protocols because their data is not encrypted and can be viewed using a network protocol sniffer. Therefore, it is undesirable to send SNMPv1 or SNMPv2 traffic across insecure networks. Allowing the SNMP traffic to cross insecure networks to reach a network you want to manage becomes challenging.

The following diagram shows that a network management system could exist on the host computer in the "Secure Central Region" to manage devices that are located in a "Secure Remote Network." To accomplish this, management traffic must flow through the "Insecure Network" region. Network managers want to avoid exposing the data inside insecure protocol packets such as SNMPv1 and SNMPv2 in this portion of the network.

SNMP Across an Insecure Network



Secure Domain Manager allows network managers to encrypt all management traffic that passes between the SpectroSERVER host and the host in the remote managed network. This lets them overcome the challenge of securely passing insecure SNMP traffic across insecure networks. This helps ensure data security when the traffic traverses the intermediate insecure networks.

Secure Domain Manager

Secure Domain Manager is a CA Spectrum network management solution that allows users to manage devices in secure networks. You can manage devices without deploying a local SpectroSERVER. Secure Domain Manager lets you manage your secure domains by securely tunneling SNMP and ICMP traffic through a secure connection. Only a single port is opened on the firewall, which allows the extended manageability without impacting security policies in place. This solution is transparent to end users and client applications, eliminating the need to perform more administrative tasks.

How Secure Domain Manager Works

Secure Domain Manager supports SNMPv1, SNMPv2, and SNMPv3 communication. It consists of two different processes, SDManager and SDConnector:

SDManager

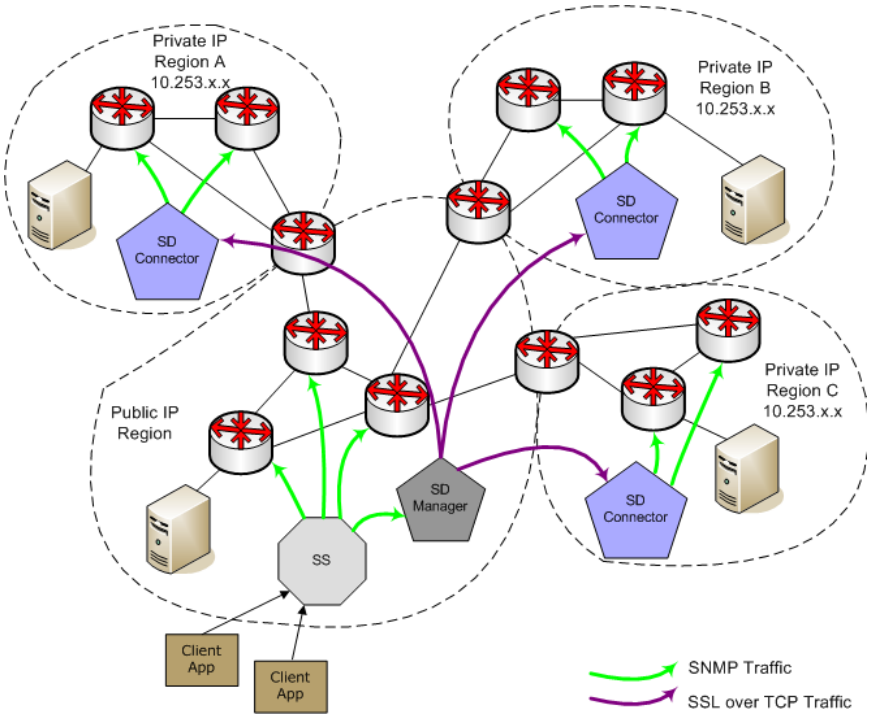
SDManager is a server messaging library that is loaded by the SpectroSERVER.

SDConnector

SDConnector is a remote process responsible for communicating with the SDManager on the SpectroSERVER. It runs on a host machine located in a remote private network and it is capable of forwarding SNMP and ICMP messages on behalf of the SpectroSERVER (which would ordinarily be deployed in the private IP region) so that it can manage devices in the private network. SDConnector is configured using a configuration file (sdc.config) which can contain both primary and backup SpectroSERVER information. It is a part of the Secure Domain Manager solution.

The following diagram shows how these processes are deployed in a secure network environment.

NAT Network Environment Using Secure Domain Manager



Note: Devices that are located in the same region as the SpectroSERVER are managed using SNMP, but without using Secure Domain Manager.

When the SpectroSERVER located in the public IP region must communicate with a device located in a remote secure region, the SpectroSERVER sends the request to the SDManager. The SDManager converts the SMNP data into a proprietary format and sends the data to the SDConnector located in the same region as the device. If the SDManager and SDConnector have been configured to run with SSL, the data is encrypted and sent through a secure tunnel to the SDConnector using SSL over TCP. When the SDConnector receives the data, it converts the data back to SNMP and sends a request to the appropriate device.

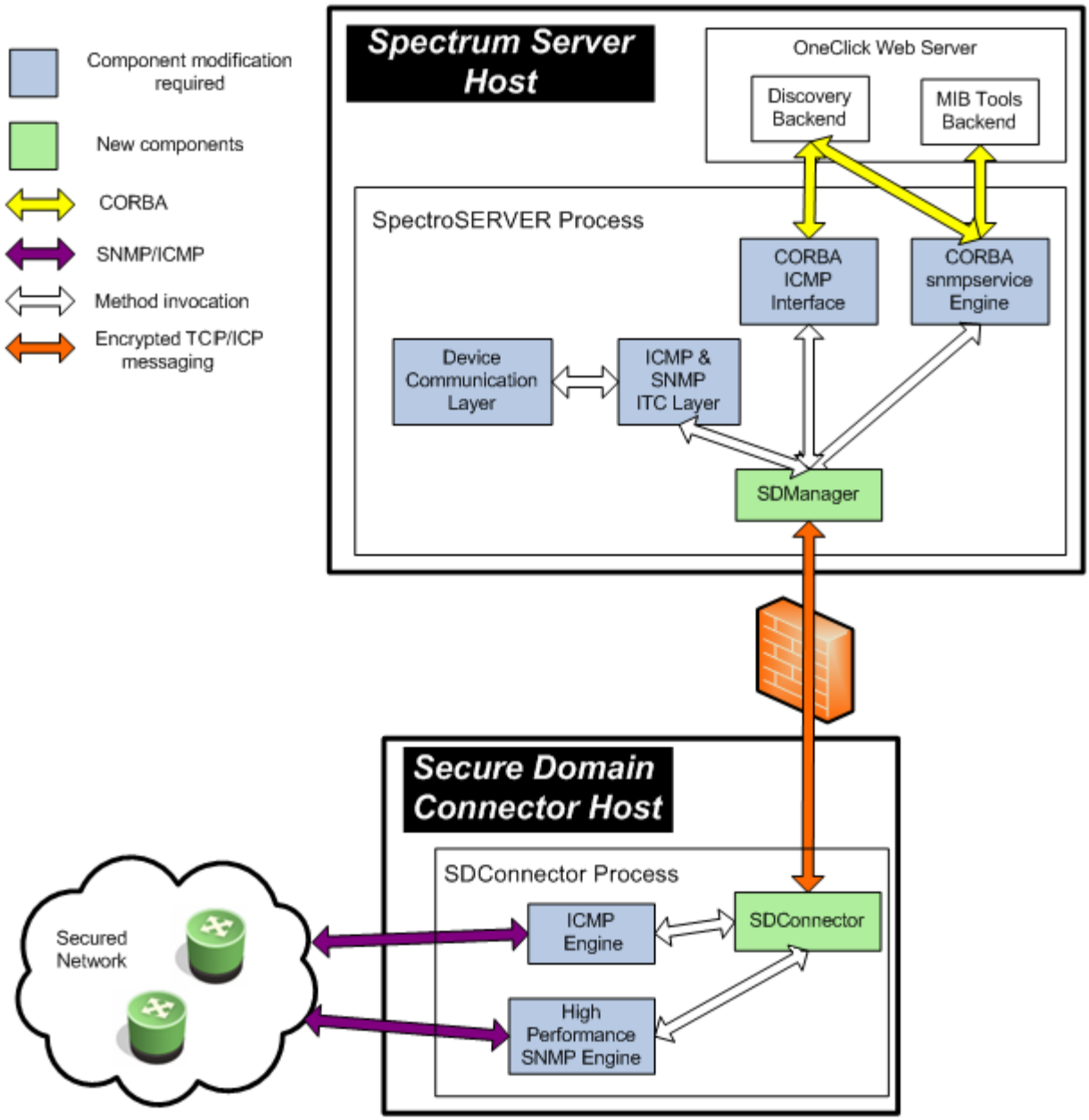
If the firewalls are deployed, it works the same way. The Network administrators must create a "hole" in each firewall that is dedicated to two well-known hosts. Devices that are located in regions that traverse more than one firewall are also manageable using this solution. To enable this communication, open a port on each firewall. The port must be a well-known port, which lets a pair of well-known hosts in adjacent regions to communicate using TCP.

When deploying Secure Domain Manager to manage overlapping IP domains, each SDConnector host machine must have a unique public IP address. The host must be able to communicate with all devices with which the SDConnector must communicate, including the SpectroSERVER host machine and all devices within the single private IP domain it manages. A likely candidate for this SDConnector host would be a machine behind the NAT that has a unique IP address that is statically assigned to it from the NAT. The SpectroSERVER uses the unique IP address of the SDConnector host machine as the additional discriminator to uniquely identify multiple devices that have the same private IP address.

Note: Certain CA Spectrum products such as Network Configuration Manager (NCM) and IP services management applications (including Multicast Manager and Enterprise VPN Manager) cannot manage overlapping IP addresses. However, you can still use Secure Domain Manager with these applications if you model their devices on the SDConnector rather than on the SpectroSERVER. Such configurations can have multiple SDConnectors deployed for each SpectroSERVER as long as the SDConnectors are not managing devices that are configured with overlapping IP addresses. Using this approach, you can still model devices on the local SpectroSERVER too, but only if they have not been configured with IP addresses that overlap with the IP addresses configured on the devices being managed through the SDConnectors.

Secure Domain Manager Architecture

The following diagram illustrates how the Secure Domain Manager operates:



The Benefits of Working with Secure Domain Manager

The Secure Domain Manager solution enhances the existing management capabilities included with CA Spectrum in the following ways:

- Lets CA Spectrum communicate with all SNMP-compliant devices: SNMPv1, SNMPv2, and SNMPv3
- Lets CA Spectrum communicate with devices located behind firewalls that block SNMP and ICMP traffic
- Simplifies firewall configuration. Because, only one hole is opened for traffic passing between two well-known hosts on a well-known port
- Lets CA Spectrum pass SNMP and ICMP traffic securely through insecure networks
- Lets CA Spectrum manage devices in overlapping IP domains (NAT environments) using a single SpectroSERVER
- Enhances the Discovery capability to discover and model devices in secure environments, one IP address space at a time

Chapter 2: Installing and Configuring Secure Domain Manager Processes

This chapter describes how to install and configure the Secure Domain Manager solution, a process that involves installing and configuring the SDConnector and SDManager.

This section contains the following topics:

How to Set Up Secure Domain Manager Processes

Setting up Secure Domain Manager involves installing and configuring Secure Domain Manager processes and then setting them up on the SpectroSERVER using OneClick.

Install and Configure Processes

Installing and configuring Secure Domain Manager processes requires the following steps:

1. Install the SDConnector process (see page 19) on a designated host.

   Note: SDManager is installed when you install the core CA Spectrum product; however, it is only active if it is included in the bundle your company purchased.

2. (Optional) Create and deploy SSL certificates (see page 21) for SSL encryption.

3. Set parameters in the configuration file for the SDConnector (see page 24) on the SDConnector host.

4. Set parameters in the configuration file for the SDManager (see page 26) on the SpectroSERVER.

Set Up Secure Domain Manager on SpectroSERVER

After you have installed and configured the SDConnector and SDManager processes, set up Secure Domain Manager on the SpectroSERVER host using OneClick. This setup involves the following steps:

1. Import the SDManager configuration file (see page 31).
2. Model an SDConnector host (see page 32).
3. Model the devices in secure domains (see page 34) that you want to manage.

Hardware Recommendations

Follow these recommendations to achieve an optimal Secure Domain Manager performance:

- To maintain an optimal SpectroSERVER modeling capacity, the SpectroSERVER/SDManager installation computer must have two CPUs: one dedicated to the SpectroSERVER and the other dedicated to servicing SDManager functions. If SDManager and SpectroSERVER are required to share a single processor to manage network elements, SpectroSERVER modeling capacity reduces by 40%.
- We recommend that you have a host computer that is dedicated solely to running each SDConnector process you deploy. The SDConnector installation system requirements are the same as those for SpectroSERVER-only installations. Except for the special requirements of multiple disk configuration.

  Note: For more information about installation requirements, see the *Installation Guide*.
- An SDConnector has only one SDManager connected to it. Two SDManagers from fault-tolerant SpectroSERVERs can be connected to a single SDConnector, if that is a requirement for your setup. See Setting Up Processes in a Fault-Tolerant Environment (see page 41) for more information.

About SDConnector CPU and Memory Usage

The SDConnector uses half the CPU capacity that the SpectroSERVER uses to manage devices. If the SpectroSERVER computer uses 50 percent total CPU and all devices are managed using SDManager, the SDConnector uses approximately 25 percent of CPU capacity, on an equally powerful system. The major difference is that the SDConnector does not use much memory. If it is devoted as an SDConnector only, 512 MB of RAM suffices, although more RAM would be better.

Install the SDConnector Process

Before using Secure Domain Manager capabilities to manage devices and applications in secure networks with CA Spectrum, install a single SDConnector process on a host computer in the secure network. Secure Domain Manager does not support running multiple SDConnector processes on the same host computer. You must be an administrative user on your Windows system, or the root user on Solaris and Linux systems when you install the SDConnector.

Note: As a best practice before upgrading the SDConnector process on any platform, stop and if required kill the process. Stopping or killing the process ensures that the process runs properly after an upgrade.

To install SDConnector

1. On the nonSpectroSERVER host machine where you want to run SDConnector, launch the appropriate CA Spectrum installer for the platform.

   Note: Only the SDConnector for the same operating environment as your SpectroSERVER is available for installation. If you have to install SDConnectors for other operating environments, contact CA Support. For information about launching the installer, see the *Installation Guide*.

   The Install dialog opens.

2. Select 'Install CA Secure Domain Connector.'

   The Introduction dialog opens.

3. Click Next to proceed.

   The License Agreement dialog opens.

4. Scroll through and read the license agreement, accept the agreement, and click Next.

   The Destination Location dialog opens.

5. Click Next to install the SDConnector in the default directory. The default directory is C:\Program Files\CA\SDMConnector on Windows and /usr/SDMConnector on Solaris and Linux.

   To install the SDConnector in a location other than the default folder, click Choose, select a folder, and click Next. The Choose button only appears for a local installation (not for a nonlocal, remote installation).

   Note: You cannot install the SDConnector into a directory that contains a space in the name.

   The Pre-Installation Summary dialog opens.

6. Click Install.

   The Installing SPECTRUM_SDM_Connector dialog opens. After the SDConnector is installed, the status changes to Install Complete and the Done button is enabled.

7. Click Done.

   The dialog closes.

8. Click Close on the initial Install dialog.

   SDConnector is installed on this host computer. The SDConnector is installed as a service, and starts automatically every time your system is restarted.

   Note: You can also check the installation log located in the directory where SDConnector was installed to verify that the installation completed successfully.

Installation Files

Be aware of the following directories and files that are created during the installation process.

On the SpectroSERVER

The CA Spectrum installation process installs the following Secure Domain Manager directories and files in the *<$SPECROOT>*/SDM directory on the SpectroSERVER:

cert

This directory is the repository for the SSL certificates you create for the SDManager.

Logs

This directory contains output logs that are generated when you import a configuration file to the SpectroSERVER. Detail of the work that is performed. The work includes any errors that occur, are contained in the log file.

README

This file provides details on how to configure Secure Domain Manager on the SpectroSERVER host.

On the SDConnector Host

The SDConnector installation process installs the following directories and files in the SDMConnector directory on the SDConnector host:

bin

This folder contains the following items for working with the SDConnector:

cert

This directory is the repository for the SSL certificates you create for the SDConnector.

README

This file provides details on how to configure the SDConnector process on the SDConnector host.

SdmConnectorService[.exe]

The executable file for the SDConnector.

Working with Certificates

Certificates are loaded by default for both SDManager and SDConnector. This lets you use the SSL encryption to secure ICMP and SNMP (SNMPv1, SNMPv2c, and SNMPv3) data that is transmitted between SDManager and SDConnector hosts across nonsecured networks. If you do not want to use the SSL encryption for any SDManager-SDConnector connections in your network environment, include the nonsecure option in the configuration files for SDManager and SDConnectors. For more information about how to use the nosecure optionsee Configure SDConnector Process Settings (see page 24).

Delete Old Certificate Files if You Are Upgrading

While upgrading to Secure Domain Manager from a version earlier to 9.*x*, delete the old certificate files. The old certificates (earlier to 9.*x*) cannot be used in this version of Secure Domain Manager. Delete the following certificate files that were installed by default in the <*$SPECROOT*>/SDM/srconf/mgr directory on SDManager hosts for earlier releases of Secure Domain Manager:

- snmpricacert.pem: Master Certificate Authority
- dsspmastercert.pem: SDManager Certificate Authority
- dsspremotecert.pem: SDConnector Certificate Authority

Create Certificates

Secure Domain Manager uses digital certificates to ensure the security. The default certificates are provided with your CA Spectrum installation and site-specific certificates can be created using the CertGen tool.

Default Certificates

If you want to use the default certificates, do not perform any actions. All default files reside in the <*$SPECROOT*>/SDM/cert directory and include the following files:

SDMCA.pem

Certificate authority. Distribute this file to any computer that uses Secure Domain Manager or Secure Domain Connector in any capacity and can be treated as a trusted CA file.

SDMCAKey.pem

Private key of CA. It can be used to issue certificates but should not necessarily be distributed to any machines.

SDMCert.p12

Application certificate that is signed by SDMCA.pem. This is the certificate file that is used between SDManager and SDConnector. It should be carefully distributed to computers that deserve trust and used to assert the identity of those computers.

CertGen[.exe]

Program that is used to generate the site-specific certificate authority, key file, and certificate file. Run CertGen -h to review all certificate options available.

openssl[.exe]

OpenSSL open source implementation of the SSL protocol.

Site-Specific Certificates

If you want to create site-specific certificates, move the default certificate files (*.pem and *.p12) to another location on the hard drive. Perform the following procedures to create and deploy the custom certificates.

Create Site-Specific Certificates

Create site-specific certificates for better security. Create these certificates on a single computer that only qualified personnel can access. This computer can be the SDManager host.

Important! You must have administrator or root privileges to create the SSL certificates for Secure Domain Manager.

Follow these steps:

1. Run the following command to create a certificate authority certificate and the private key for the certificate authority certificate:

   ```
   CertGen -t ca -c US
   ```

   You only have to perform this step once to create the necessary certificate authority certificate for your organization.

   The following files are created:

   SDMCA.pem

   SDMCAKey.pem

   Note: The default certificate authority and key file that come with Secure Domain Manager are read-only files. If you receive a permission error, check your user privileges or move SDMCA.pem and SDMCAKey.pem to another location and run the command again.

2. Run the following command to create a certificate for the SDManager:

   ```
   CertGen -t cert -c <Country Code>
   ```

   The SDMCert.01.p12 file is created.

3. (Optional) For the added security, use the -p option to generate the certificate with a password as follows:

   ```
   CertGen -t cert -p <password> -c <Country Code>
   ```

   Enter the password in the sdc.config file and sdm.config file.

4. Rename SDMCert.01.p12 to SDMCert.p12.

   The new site-specific certificate is ready for use.

Deploy Site-Specific Certificates

After you create your certificate files, perform the following tasks:

- Deploy the certificate files on the SDManager hosts and on the SDConnector hosts.
- Restart the SpectroSERVER on the SDManager hosts and the SDConnector process on the SDC hosts.

To deploy certificates, copy the SDMCA.pem file that you created to the *<$SPECROOT>*/SDM/cert directory on the SDManager host computer and to the cert directory under the SDConnector installation on the SDConnector hosts that will connect to the SDManager host. Administrator, or root should own the SDMCert.p12 file.

Important! Retain the SDMCAKey.pem file on the computer where you plan to create more certificates. Restrict the file to authorized personnel only. This computer can be the SDManager host computer but is not a requirement.

After the certificates have been deployed, restart both the SpectroSERVER on the SDManager hosts and the SDConnector process on the SDC hosts. For information on restarting the SDConnector Process, see Start, Stop, and Restart the SDConnector Process on Windows (see page 29) or Start, Stop, and Restart the SDConnector Process on Solaris and Linux (see page 29).

Configure SDConnector Process Settings

This section describes the configuration options you can set in the SDConnector configuration file (sdc.config). This configuration file is read at startup and the options specified are applied at that time. Only one line of options is accepted in the sdc.config. The following is a sample line from an sdc.config file. It specifies that the SDConnector will accept connections from SDManager (192.168.0.2):

```
-accept 192.168.0.2
```

To configure SDConnector settings

1. Create (or open, if it already exists) a file named 'sdc.config' in the SDMConnector\bin directory on the SDConnector host machine using a text editor.
2. Add and specify details for the following options on one line in the file, according to your particular requirements:

   -accept remote_ipaddr:[local_port]

   Accepts a connection from an SDManager running on a host at address *<ip>* at local port number *<port>*. Connections must originate from the IP address specified; otherwise, connection attempts are disregarded.

   If this option is specified, the SDManager that connects to this SDConnector must have the -remoteconnect option that specifies this SDConnector *<ip>* in its configuration file (sdm.config). Also, if this option is specified you cannot connect (-connect) to that SDManager.

   -buffersize *<size>*

   Specifies the size of the send and receive socket buffer sizes in bytes.

   Default: 262,144 (256k, which should be sufficient in most deployments)

-certdir <*dir*>

Specifies the directory for SSL certificates (application certificate, private key, and the certificate authority certificate) if they are not located in the default directory (/cert).

If the -nosecure option is specified, certificates are not accessed.

-certpassword <passwd>

Provides the certificate password. If you are using the default certificates that ship with Secure Domain Manager, then -certpassword need not be supplied. Otherwise, supply the certificate password using this option. If the password contains spaces, it must be enclosed in quotation marks ("). CA Spectrum assumes that the password for the application certificate will be encrypted.

Note: If you use -certpassword, it must be the first option declared in the config file.

-connect remote_ipaddr:[remote_port]

Connects to the SDManager running on a host at IP address <*ip*> and port <*port*>. If <*port*> is not specified, 6844 is assumed.

If this option is specified, the SDManager to which this SDConnector connects must have the -remoteaccept option that specifies this SDConnector's IP address in its configuration file (sdm.config).

If this option is specified, this SDConnector cannot accept (-accept) connections from or listen (-listen) for connections from the specified SDManager (sdm.config).

-keepalive <*n*>

Changes the default internal timeout (in seconds) when the SDManager or SDConnector sends out a small message to verify the network connection is still alive. If either the SDManager or an SDConnector does not hear from the other within three times the value of <*n*>, the connection is terminated.

Default: 10 seconds

-listen [port]

By default, the SDConnector listens at port 6844 for connection requests from any SDManagers. However, if any -connect or -accept options are specified, then the SDConnector no longer listens by default.

A port specified in a -listen option trumps a port specified in an -accept option. That is, if a port is specified in a -listen option, there will be no verification done of the source IP address for that port.

Note: -listen and -listen6 are mutually exclusive.

-listen6 [local_port]

Accept connections from any IPv6 SDManager on the given port.

Note: -listen and -listen6 are mutually exclusive.

-loglevel fatal|error|warning|info|debug

Specifies the types of messages to log.

Default: warning (includes error and fatal as well)

-maxlogsize *<n>*

Sets the maximum sdmLog.log size in megabytes.

Default: 5M

Minimum: 1M

-nosecure

Disables Secure Socket Layers (SSL) security, which is enabled by default. If the -nosecure option is used before any -connect or -accept entries, SSL is disabled for all connections. Otherwise, you can specify the -nosecure option after each -connect or -accept entry and it will pertain just to that entry.

If SSL security is requested, the data stream is encrypted, and mutual cryptographic authentication is enforced. If either the SDManager or the SDConnector requests security, then security is mandatory on that connection.

-trappoll *<n>*

Forward traps to the SDManager every *<n>* seconds.

Default: 15 seconds

-withfips

Specifies to run with FIPS mode. FIPS mode is off by default.

Note: If an empty sdc.config is created, SDConnector listens for connections from any SDManager on port 6844; the SDManager initiates the connection.

3. Save and exit the file.

   The SDConnector is configured.

   Note: You must restart the SDConnector process every time you make updates to the sdc.config file.

Configure SDManager Process Settings

The SDManager configuration file (sdm.config) specifies the operational settings for the SDManager process. By default, the SDManager process is disabled. The SDManager process will not work until you create the sdm.config file and configure it according to your needs. After you configure the sdm.config file for the first time, or any time you revise its settings, you must import it into CA Spectrum to put SDManager settings into effect on the SpectroSERVER. Fore more informatio, see Import the SDManager Configuration File (see page 31). You can configure the sdm.config either before or after the SpectroSERVER is started.

Only one line of options is accepted in the sdm.config. The following is a sample line of options for an sdm.config. It is specifying the connections (-remoteconnect) to two SDConnectors (172.24.148.196 and 172.19.32.199):

```
-remoteconnect 172.24.148.196 -remoteconnect 172.19.32.199
```

Note: If you use the -nosecure option to launch one or more of the SDConnector processes, you must specify the same -nosecure option for the corresponding -remoteconnect/-remoteaccept entry in the SDManager options, or simply specify -nosecure before all -remoteconnect/-remoteaccept entries to disable SSL for all connections.

To configure SDManager settings

1. Create (or open, if it already exists) a file named 'sdm.config' in the *<SPECROOT>*\SDM directory on the SpectroSERVER host machine using a text editor.
2. Add and specify details for the following options on one line in the file, according to your particular requirements:

   -apiclientport [port]

   Sets the port to listen for API client connections. This parameter applies to the stand-alone SDManager process only.

   -buffersize *<size>*

   Specifies the size of the send and receive socket buffer sizes in bytes.

   Default: 262,144 (256k, which should be sufficient in most deployments)

   -certdir *<dir>*

   Specifies the directory for SSL certificates (application certificate, private key, and the certificate authority certificate) if they are not located in the default directory (/cert).

   If the -nosecure option is specified, certificates are not accessed.

   -certpassword <passwd>

   Provides the certificate password. If you are using the default certificates that ship with Secure Domain Manager, then -certpassword need not be supplied. Otherwise, supply the certificate password using this option. If the password contains spaces, it must be enclosed in quotation marks ("). CA Spectrum assumes that the password for the application certificate will be encrypted.

   Note: If you use -certpassword, it must be the first option declared in the config file.

   -clientServiceThreads *<n>*

   Sets the number of threads per client that will process requests. This parameter applies to the stand-alone SDManager process only.

-keepalive *<n>*

Changes the default internal timeout (in seconds) when the SDManager or SDConnector sends out a small message to verify the network connection is still alive.

Default: 10 seconds

If either the SDManager or an SDConnector does not hear from the other within three times the value of *<n>*, the connection is terminated.

-loglevel fatal|error|warning|info|debug

Specifies the types of messages to log.

Default: warning (includes error and fatal as well)

-maxapiconnections *<n>*

Sets the maximum number of API client connections to *<n>*. This parameter applies to the stand-alone SDManager process only.

-maxlogsize *<n>*

Sets the maximum sdmLog.log size in megabytes.

Default: 5M

Minimum: 1M

-nosecure

Disables the Secure Socket Layers (SSL) functionality, which is enabled by default. If the -nosecure option is used before any -remoteconnect or -remoteaccept entries, SSL is disabled for all connections. Otherwise, you can specify the -nosecure option after each -remoteconnect or -remoteaccept entry and it will pertain just to that entry.

If SSL security is requested, the data stream is encrypted, and mutual cryptographic authentication is enforced. If either the SDManager or the SDConnector requests security, then security is mandatory on that connection.

-remoteaccept (-rema) remote_ipaddr[:local_port]

Accepts a connection from an SDConnector running on a host at address *<ip>* at local port number *<port>*. You must specify the SDConnector's public IP address.

If this option is specified, the SDConnector that connects to this SDManager must have the -connect option that specifies this SDManager's IP address in its configuration file (sdc.config). Also, if this option is specified, you cannot connect (-remoteconnect) to the SDConnector (sdc.config).

-remotebackup (-remb) remote_ipaddr[:remote_port]

Specifies the backup SDConnector in a fault-tolerant Secure Domain Manager setup using the SDConnector's public IP address. For more information, sSee Stting Up Processes in a Fault-Tolerant Environment (see page 41).

-remoteconnect (-remc) remote_ipaddr[:remote_port]

Connects to the SDConnector running on a host at IP address *<ip>* and *<port>*. If *<port>* is not specified, 6844 is assumed. You must specify the SDConnector's public IP address.

If this option is specified, the SDConnector to which this SDManager connects must have the -accept option that specifies this SDManager or the -listen option in its configuration file (sdc.config). Also, if this option is specified you cannot accept connections (-remoteaccept) from the specified SDConnector in this configuration file.

-withfips

Specifies to run with FIPS mode. FIPS mode is off by default. If changing configuration from FIPS mode to non-FIPS, or vice versa, you must restart the application.

Note: If the sdm.config file is empty, the SDManager process is disabled.

3. Save and close the sdm.config file.

   The SDManager is configured.

More information:

Import the SDManager Configuration File (see page 31)

Start, Stop, and Restart the SDConnector Process on Windows

Use the Services manager to start, stop, or restart the SDConnector process. The SDConnector process is listed under the name 'Secure Domain Connector.'

Start, Stop, and Restart the SDConnector Process on Solaris and Linux

To start the SDConnector process, log in as root, open a command line console, and enter the following commands:

```
$ cd /etc/init.d

$ ./sdmconnector start
```

To stop the SDConnector process, issue the **./sdmconnector stop** command.

To restart the SDConnector process, issue the **./sdmconnector restart** command.

Chapter 3: Working with Secure Domain Manager

This chapter describes how to import the SDManager configuration file (sdm.config) into CA Spectrum and model SDConnector hosts and devices in secure domains. This chapter also describes OneClick tools that are use to locate Secure Domain Manager components. These components are used to ping devices in a secure domain. Ping the devices to view device MIBs, and to view information about the SDManager and SDConnector models.

This section contains the following topics:

Import the SDManager Configuration File

Import the sdm.config file into CA Spectrum before you can begin using OneClick with the Secure Domain Manager product and whenever you want to update the SDManager configuration. See Configure SDManager Process Settings (see page 26) for information about setting sdm.config parameters.

Note: You can import the SDManager configuration file before or after you create models for SDConnector hosts. If, however, you import an sdm.config file before you create models for SDConnector hosts, CA Spectrum automatically models the hosts as SDConnectorProcess model types. See Model SDConnector Hosts (see page 32) for more information about modeling options, including how to model SDConnectors as Pingable and Host_Device model types.

To import the SDManager configuration file

1. Click Secure Domain Manager in the Navigation panel in the OneClick Console.
2. Click the Information tab in the Component Detail panel and expand the Configuration subview.
3. Click Import.

   The Import Secure Domain Manager Configuration confirmation dialog opens.

4. Click Yes to confirm that you want to import the SDManager configuration file (sdm.config).

   The Import Secure Domain Manager Configuration dialog indicates whether the import started successfully. The dialog also provides information to check the output log to determine whether the import worked. The import log file in the SDM/Logs directory provides the troubleshooting information. This information is used to fix errors after an unsuccessful import.

5. Click OK.

   If the configuration file has been imported correctly, the Secure Domain Manager Status field displays "Configured." If an sdm.config file which contains no arguments to define how connections between SDManager and SDConnectors are established is imported, SDManager is disabled and the Secure Domain Manager Status field displays "Not Configured."

   Note: If the sdm.config file is edited while the SpectroSERVER is not running, the SpectroSERVER automatically imports the new sdm.config file when it is started. You can verify whether the import was successful by checking the latest log file.

More information:

Configure SDManager Process Settings (see page 26)
SDManager Model Information View (see page 38)

Model SDConnector Hosts

Use the Model by Type option in the OneClick Topology view to model an SDConnector host computer as one of the three following model types:

SDConnectorProcess

The SDConnectorProcess model type is the default model type for SDConnectors. This model type does not allow you to manage the device status, but it does allow you to see the host computer that is represented in the OneClick Secure Domain Manager model hierarchy and provides access to the views discussed in SDConnector Model Information View (see page 40).

Note: Use meaningful names for SDConnector host models that clearly identify the hosts. The model names appear in the Secure Domain Manager views in OneClick.

Host_Device

Use the Host_Device model type if the host computer is running an SNMP agent.

Pingable

Use the Pingable model type if the host computer only supports ICMP.

If you use either the Host_Device or Pingable model type, you can monitor the status of the host computer. See SDConnector Modeling and CA Spectrum Fault Isolation (see page 33) for information about leveraging the CA Spectrum fault isolation capabilities by modeling SDConnector hosts as Host_Device or Pingable models.

SDConnector Modeling Considerations

- By default, CA Spectrum automatically models an SDConnector host computer as an SDConnectorProcess model type if you have not created models for the computer before you initially import the SDManager configuration file.
- If you prefer to model hosts as Pingables, or Host_Devices, model the hosts as your preferred type before the import. Alternately, destroy the SDConnectorProcess models after the import. Then, model the hosts as Pingables, or Host_Devices.

  Note: If you use the Model By IP option to create a model representing the SDConnector host without first destroying the existing SDConnectorProcess model, CA Spectrum copies and pastes the SDConnectorProcess model into the topology view from which the Model By IP option was invoked.
- Destroying the SDConnector host model in OneClick does not prevent the CA Spectrum from using the actual SDConnector for the device communication. The SDConnector can only be destroyed by reimporting the SDManager configuration (after editing the sdm.config file to remove the SDConnector).
- If you accidentally destroy an SDConnectorProcess model, CA Spectrum recreates the model the next time you import the SDManager configuration file. If you destroy a Pingable or Host_Device model, CA Spectrum creates an SDConnectorProcess model the next time you import the SDManager configuration file. If you want to restore your Pingable or Host_Device model, explicitly recreate the model and then import the configuration file.

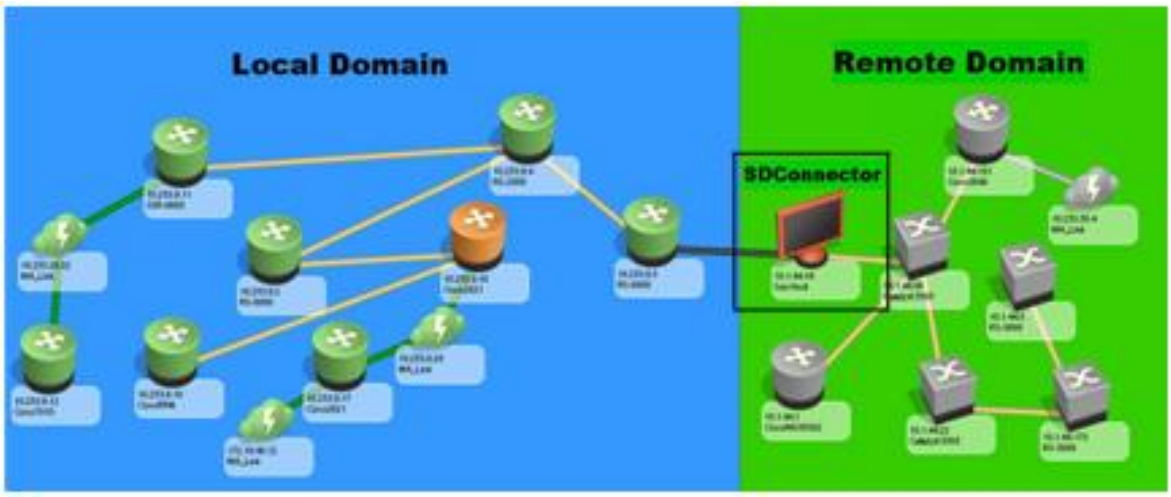
SDConnector Modeling and CA Spectrum Fault Isolation

As described in Model SDConnector Hosts (see page 32), when you model SDConnectors, you can choose one of the following model types:

- SDConnectorProcess
- Host_Device
- Pingable

We recommend you to model the SDConnector host as a Host_Device or Pingable model type. This model type lets CA Spectrum fault isolation to work correctly when a remote SDConnector process goes down, or loses its connection. CA Spectrum isolates the cause of an outage to the SDConnector host model fully, virtually eliminating unresolved fault alarms.

Although the SDConnector host is mostly connected to a switch on the edge of a network. Logically, it is the bridge between the public domain and secure domain regions and it must be modeled accordingly. Place the SDConnector host model between the two models for the devices that are routing traffic between the public domain and secure domain regions. The following diagram illustrates this connection, showing the SDConnector as a Host_Device model.



Model Devices in Secure Network Domains

After you model an SDConnector host, model the network devices that you want to manage in the secure domain where the SDConnector host is located. Model the network devices one at a time using the OneClick Create Model By IP option, or Discovery. You can place the models anywhere in the Topology view. After you have successfully created models, CA Spectrum can communicate with them using the SDConnector process.

Create Model by IP

Use the OneClick Create Model By IP option to model each device in a secure domain.

Note: For more information about modeling in OneClick, see the *Modeling and Managing Your IT Infrastructure Administrator Guide*.

To model a device in a secure domain using the Model by IP option

1. Click the Model by IP option in the Topology view.

   The 'Create Model by IP Address' dialog opens.

2. Type the network address of the device you want to model in the Network Address field.

3. From the Secure Domain drop-down list, select the IP address of the host running the SDConnector or the name that has been configured for the SDConnector host in the secure domain where the device you are modeling is located.

   Note: You can provide a secure domain name for the SDConnector host by changing the host model name in OneClick. See SDManager Model Information View (see page 38) for information about enabling the secure domain name as a selection option.

4. Select the SNMP version compatible with the device you want to manage in the SNMP Communications Options section.

5. Click OK.

Discovery

Use OneClick Discovery to discover and model all devices in a secure domain with an SDConnector host. Keep in mind the following points while discovering devices with overlapping IP addresses:

- Only one SDConnector can be used for each Discovery.
- Although you can use Layer 2 mapping, its effectiveness is dependent upon the accuracy of the Source Address and the Spanning Tree tables.
- The Protocol Options settings:
  - Do not use Layer 3 Autodiscovery mapping. Deselect IP Address Tables and IP Route Tables in the Protocol Options dialog.
  - Do not use Proprietary Discovery Protocols in Cisco, or the Nortel environments because they use IP addresses to convey neighbor relationships. Deselect Proprietary Discovery Tables in the Protocol Options dialog.
  - Do not use the Pingable mapping. Deselect ARP tables for Pingables in the Protocol Options dialog.

Discover Devices Using an SDConnector Host

Follow these steps:

1. Click Tools, Utilities, Discovery Console from the main menu.

   The Discovery Console opens.

2. Complete the Discovery configuration for the secure domain from which you want to model devices.

   Note: For more information about configuring Discovery, see the *Modeling and Managing Your IT Infrastructure Administrator Guide*.

3. Click Advanced Options in the Configuration tab.

   The Advanced Options dialog opens.

4. In the Discovery Options section, from the Secure Domain drop-down list, select the IP address of the host running the SDConnector in this secure domain or the name that is specified for the secure domain.

   Note: You can provide a secure domain name for the SDConnector host by changing the host model name in OneClick. See SDManager Model Information View (see page 38) for information about enabling the secure domain name to be a selection option.

5. Click OK.

   The Advanced Options dialog closes and your changes are saved.

6. Click Discover in the Discovery Console.

   The Discovery that you configured runs. After Discovery, view all the devices that are listed in the Secure Domain Connector Device Table of its corresponding SDConnector host icon.

   Note: If you have already modeled the host machine running a remote SDConnector process using the SDConnectorProcess model and you perform a Discovery on the network region where the host exists, Discovery may create an additional model of the host using the Host_Device, or the Pingable model. Delete this duplicate model once it has been created or you can filter this model out of the Discovery result that is set before it is created.

About Maintaining Device Secure Domain Membership

In a NAT environment, multiple SDConnectors are used to manage the same IP ranges. When the duplicate IP ranges exist, CA Spectrum cannot determine the SDConnector that must manage each device. So, specify this information.

When discovering or modeling new devices in CA Spectrum, you can set the secure domain using the OneClick Model by IP view or OneClick Discovery. To update the secure domain for an existing device model, use the OneClick Attribute Editor to edit the Secure Domain Address attribute. This automatically updates the Secure Domain Name. When a new SDManager configuration file (sdm.config) is imported into CA Spectrum, any existing devices that were assigned to an old secure domain will still be assigned to it. Red alarms will likely be generated on these models.

Access Secure Domain Manager Searches

OneClick includes various predefined Secure Domain Manager search options.

To access Secure Domain Manager search options, expand the Secure Domain Manager folder in the Locater tab in the OneClick Console.

The predefined Secure Domain Manager searches available to you are displayed.

Check Device Accessibility in a Secure Domain

Determine whether the devices are accessible by using the OneClick Ping menu option to ping devices that are located in a secure domain.

Note: A successful ping does not display the number of bytes returned by the pinged device in a secure domain.

To check the device accessibility in a secure domain, right-click the device for which you want to assess accessibility in the OneClick Console and click Ping.

The Ping dialog opens, listing the results of the ping request. For example:

```
Secure reply from 10.254.1.5: icmp_seq=4. time =140. ms
```

If this device was not in a secure domain, the result would appear as follows:

```
64 bytes from 10.254.1.5: icmp_seq=4. time =140. ms
```

View a Device MIB in a Secure Domain

View a device MIB in a secure domain with MIB Tools. First, specify the SDConnector for the secure domain where the device is located. The following procedure describes how to specify an SDConnector.

Note: For more information about using MIB Tools, see the *Certification User Guide*.

Follow these steps:

1. Select the device that you want to investigate with MIB Tools.
2. Right-click the device and select Utilities, MIB Tools.

   MIB Tools open. The Contact Criteria is prepopulated with the selected SNMP contact information of the device. MIB Tools attempts to contact the device.

   If MIB Tools *cannot* contact the device an error message appears and the Contact Status indicator turns red.

   If MIB Tools *can* contact the device, the Contact Status indicator turns green.

   A status dialog also appears which shows the progress of retrieving and loading the MIB Tools database.
3. Click Advanced Options in the Contact Criteria section.

   The MIB Tools: Advanced Options dialog appears.
4. Select the applicable secure domain from the Secure Domain drop-down list.
5. Click OK.

   The Advanced Options dialog closes and your changes are saved.
6. Click Contact in the Contact Criteria section and verify that MIB Tools can contact the device successfully.
7. Close MIB Tools.

   MIB Tools closes and you have specified the SDConnector for the device.

SDManager Model Information View

The Information tab in the Component Detail panel provides information about and configuration controls for the selected SDManager model in the following sections:

General Information

The General Information section provides standard information about the Secure Domain Manager model such as its model class and security string.

Configuration

The Configuration section includes the following content:

Import

Imports the SDManager configuration file (sdm.config) into CA Spectrum.

Secure Domain Manager Status

Indicates the configuration status of the SDManager as follows:

- **Configured:** Indicates that the file has been successfully imported.
- **Not Configured:** Indicates either that a custom or edited sdm.config file has never been imported, an sdm.config file with no arguments has been imported, or an sdm.config file that contains errors has been imported.

Secure Domain Display Option

Specifies whether CA Spectrum displays the name that is used to identify the SDConnector host (and its domain) or the SDConnector host IP address. You can choose either "Display Secure Domain Name" or "Display Secure Domain Address" from the drop-down list. This determines which the type of SDConnector identifier is used throughout all OneClick views.

Local Domain

Specifies the text that appears in the Secure Domain column for locally managed models (models that are not included in a secure domain).

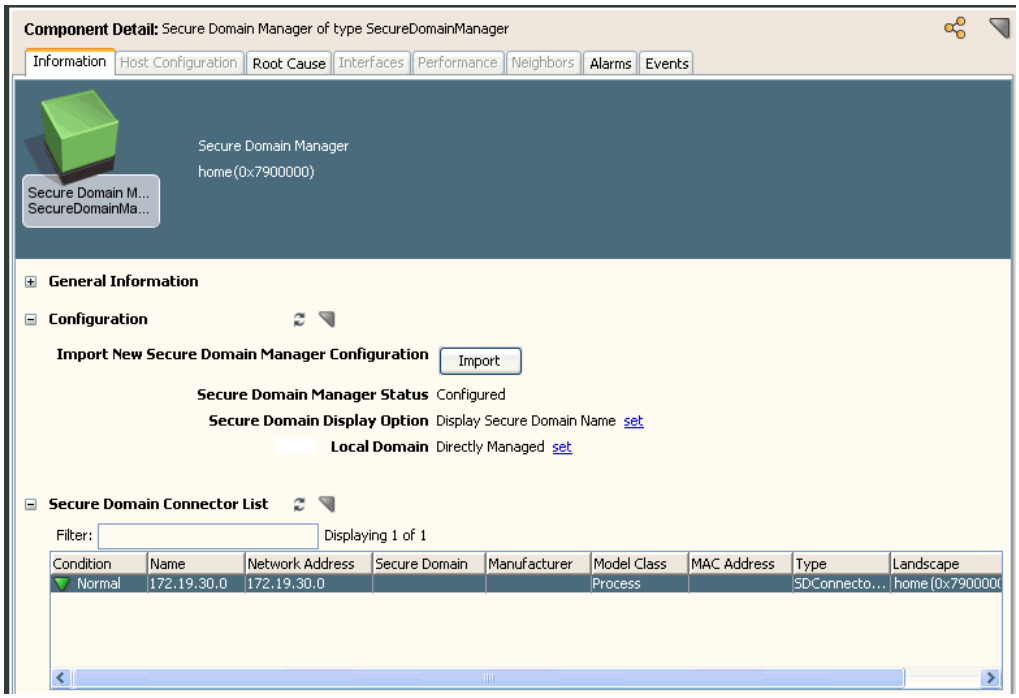
Default: Directly Managed

Note: The Secure Domain column appears in OneClick list views only when Secure Domain Manager is installed.

Secure Domain Connector List

Displays all the host machines currently running SDConnector processes in remote network regions.

The following image shows an example of the Component Detail panel for a selected SDManager model:



More information:

Import the SDManager Configuration File (see page 31)

SDConnector Model Information View

The Information tab that is located in the Component Detail panel provides information about the selected SDConnector. The General Information and the SPECTRUM Modeling Information categories provide standard information about the SDConnector model. There is also a Secure Domain Connector section that includes the following subsection:

Secure Domain Connector Device Table

The Secure Domain Connector Device Table lists all the devices that are managed by the selected SDConnector. It also lets you print, export, and filter the list of devices. You can click the Name hyperlink of a device in this list to navigate directly to that device in the Topology view.

Chapter 4: Setting Up Processes in a Fault-Tolerant Environment

This chapter describes how to set up SDConnectors to connect to SDManagers on primary and backup SpectroSERVERs in a fault-tolerant SpectroSERVER environment. This chapter also describes how to set up primary and backup SDConnectors.

This section contains the following topics:

Set Up SDManager in a Fault-Tolerant SpectroSERVER Environment

In a fault-tolerant SpectroSERVER environment, install the SDManager on both the primary SpectroSERVER and the backup SpectroSERVER. Each SDConnector that communicates with this SDManager is configured to connect to the primary and backup SpectroSERVERS. If the primary SpectroSERVER fails, the backup SpectroSERVER takes over communications with each SDConnector.

Follow these steps:

1. Deploy an SDConnector on each secure domain that you want to manage.

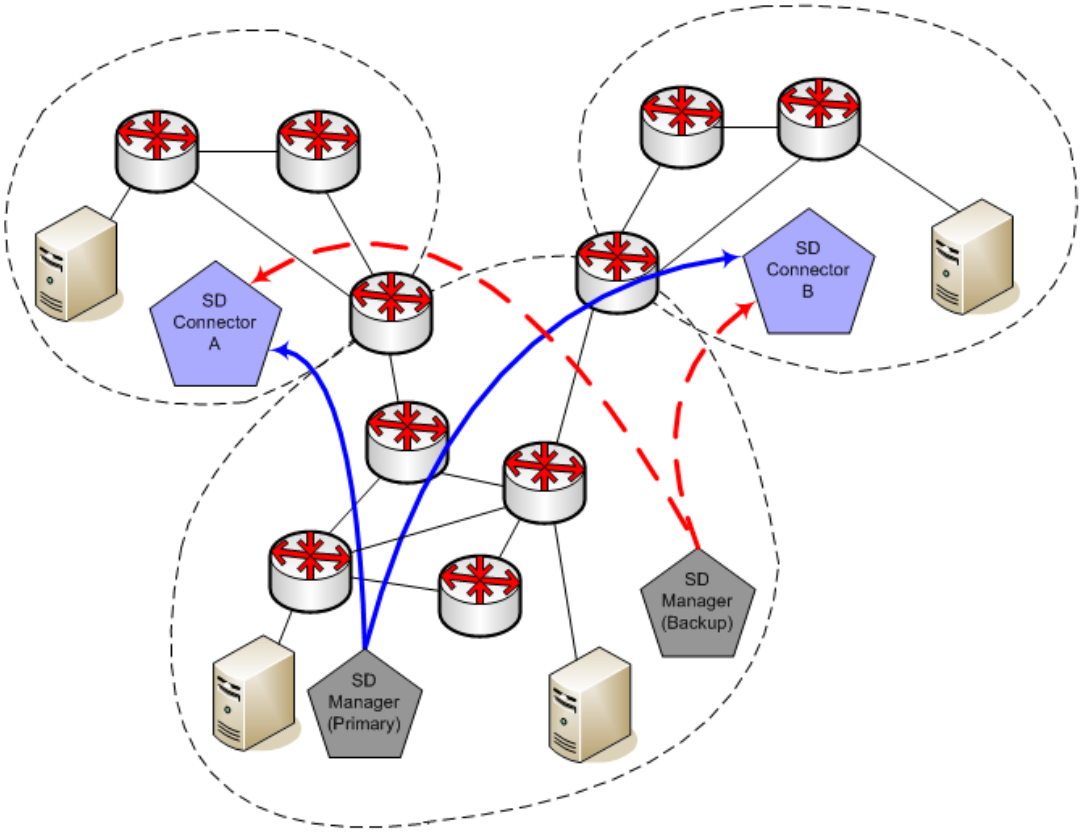
   Note: See Installing and Configuring Secure Domain Manager Processes (see page 17) for detailed instructions about how to deploy SDConnectors.

2. Configure each SDConnector to accept connections from both the primary SpectroSERVER and the backup SpectroSERVER. For example, 172.24.1.2 and 172.24.3.4 respectively:

   ```
   -accept 172.24.1.2 -accept 172.24.3.4
   ```

Fault-Tolerant SpectroSERVERs (SDManagers)

The following diagram shows how two SDConnectors can be connected to both a primary SDManager and a backup SDManager:



Configuration setting for both SDManagers in the sdm.config:

```
-remoteconnect <IP of SDConnector A> -remoteconnect <IP of SDConnector B>
```

Configuration setting for both SDConnectors in the sdc.config:

```
-accept <IP of primary SDManager> -accept <IP of backup SDManager>
```

Set Up Fault-Tolerant SDConnectors

Secure Domain Manager supports the Backup functionality on a per-SDConnector basis. A backup SDConnector must be able to manage all the devices the primary SDConnector does, not merely a subset of them.

When you import a backup configuration into CA Spectrum, the backup SDConnector is not automatically modeled. If the primary SDConnector goes down, the backup functionality takes over *transparently*. There is no visible indication that the primary SDConnector is down. Also, because backups are not modeled they do not appear in the OneClick Console Model by IP or Discovery Configuration views or in MIB Tools.

Follow these steps:

1. Deploy both a primary and a backup SDConnector for each remote domain that you want to manage.

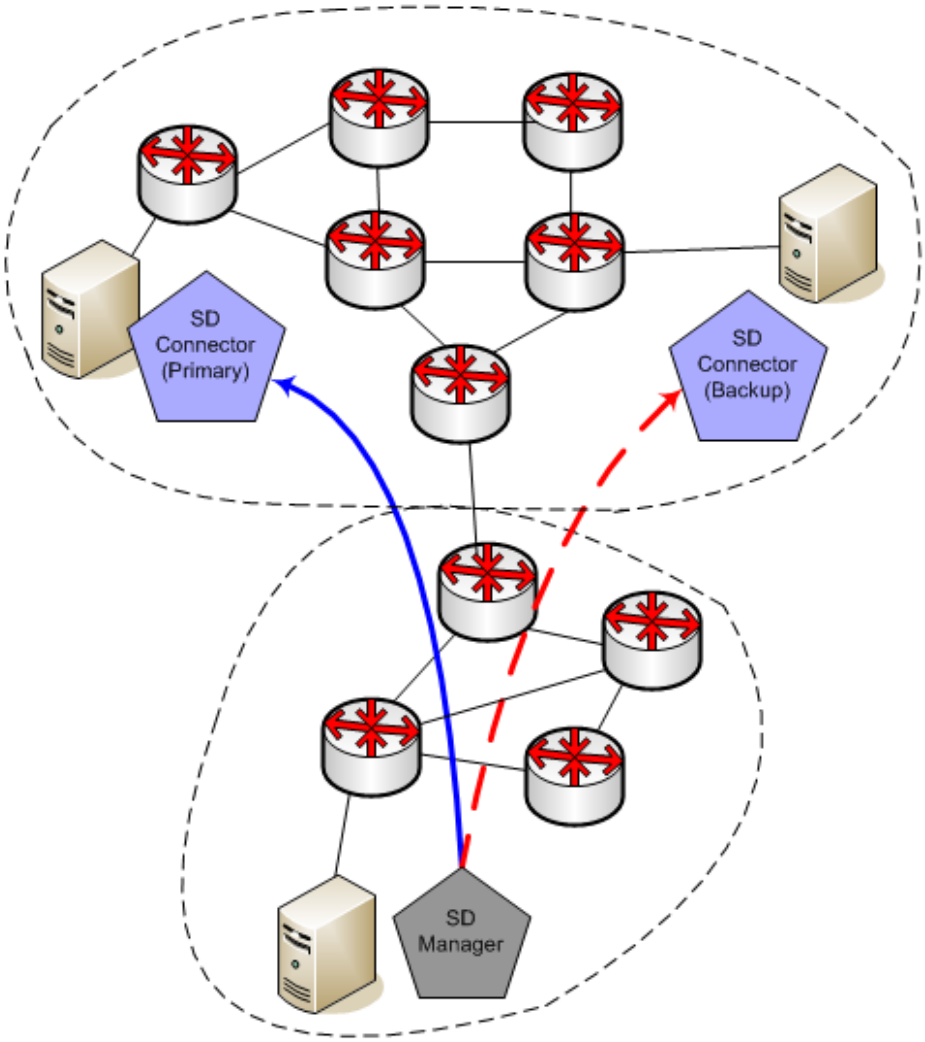
   Note: See Installing and Configuring Secure Domain Manager Processes (see page 17) for detailed instructions about how to deploy SDConnectors.

2. Configure the SDManager to connect to the primary and backup SDConnectors by modifying the sdm.config file as shown in the following example:

   ```
   -remoteconnect <IP of primary SDC> -remotebackup <IP of backup SDC>
   ```

Fault-Tolerant SDConnectors

The following diagram depicts two SDConnectors connected to a single SDManager:



Configuration setting for the SDManager in the sdm.config:

```
-remoteconnect <IP of primary SDConnector> -remotebackup <IP of backup SDConnector>
```

Configuration setting for both SDConnectors in the sdc.config:

```
-accept <IP of SDManager>
```

Appendix A: Troubleshooting Secure Domain Manager

This section describes some potential Secure Domain Manager problems and their solutions.

This section contains the following topics:

Error Messages

This section provides information about Secure Domain Manager error messages. SDManager errors appear in the SDManager.out file; SDConnector errors appear on the terminal display.

Certificate Not Valid Error

Valid on Linux, Solaris, and Windows

Symptom:

The following SDConnector error message appears when a mismatch is found in certificates or security settings:

```
SdmEtpkiConnectEndpoint run() invalid socket security. No connection attempts will be made to the host.

Please verify that certificates and security configurations are correct.
```

Solution:

Verify that machines on which SSL is deployed have matching certificates.

Port Conflicts

SDConnector Requires a Custom SNMP Trap Port

Valid on Linux, Solaris, and Windows

If there exists a need to change the trap port that the SDConnector listens for the SNMP traps on, configure a custom listening port.

Note: In the following procedure, port 951 is used as an example of a new custom listening port.

Follow these steps:

1. Configure SDConnector to listen for traps on a custom port by modifying the sdc.rc file as follows:

   ```
   snmp_trap_port = 951
   ```

2. Restart SDConnector process by rebooting the computer.

   The SDConnector now listens traps on port 951.

Installation Issues

When on some Windows installations the SDConnector service is not installed. Or, when it is installed, but not started. Then, manually install SDConnector service on Windows.

Follow these steps:

1. From a command prompt, navigate to the following folder:

   <SDC Install directory>/bin

2. Run the following command:

   ```
   SdmConnectorService.exe --install
   ```

3. Start the service from the Services window or run the following command:

   ```
   SdmConnectorService.exe --start
   ```

Index