CHAPTER 16

Configuring Switch Security

The authentication, authorization, and accounting (AAA) strategy verifies the identity of, grant access to, and tracks the actions of remote users in all switches in the Cisco MDS 9000 Family. The Remote Access Dial-In User Service (RADIUS) and Terminal Access Controller Access Control System Plus (TACACS+) provide AAA solutions.

Based on the user ID and password combination provided, switches perform local authentication using a local database or remote authentication using AAA server(s). A global, preshared, secret key authenticates communication between the AAA servers. This secret key can be configured for all AAA server groups or for only a specific AAA server. This kind of authentication provides a central configuration management capability.

This chapter includes the following sections:

Switch Management Security

Management security in any switch in the Cisco MDS 9000 Family is implemented using the command-line interface (CLI) or Simple Network Management Protocol (SNMP).

CLI Security Options

You can access the CLI using the console (serial connection), Telnet, or Secure Shell (SSH). For each management path (console or Telnet and SSH), you can configure one or more of the following security control options: local, remote (RADIUS or TACACS+), or none.

- Remote security control
  - Using Remote Authentication Dial-In User Services (RADIUS). See the "Configuring RADIUS" section on page 16-5.
  - Using Terminal Access Controller Access Control System plus (TACACS+). See the "Configuring TACACS+" section on page 16-10.
- Local security control. See the "Local AAA" section on page 16-15.
- Trivial authentication. See the "No AAA Authentication" section on page 16-15.

These authentication mechanisms can also be used to configure AAA for the following scenarios:

- iSCSI authentication (see the "Authentication Mechanism" section on page 22-65).
- Fibre Channel Security Protocol (FC-SP) authentication (see the Chapter 17, "Configuring Fabric Security.")

SNMP Security Options

The SNMP agent supports security features for SNMPv1,SNMPv 2c, and SNMPv3. Normal SNMP security mechanisms apply to all applications that use SNMP (for example, Cisco MDS 9000 Fabric Manager).

See the "SNMP Security" section on page 16-30.

Refer to the *Cisco MDS 9000 Family Fabric Manager Configuration Guide* for information on the Fabric Manager.

Note Users and roles configured through the CLI are different from users and roles configured through SNMP. These configurations do not directly correspond with each other. However, you can configure both CLI and SNMP identically, if required.

Switch AAA Functionalities

Using the CLI, you can configure authentication, authorization, and accounting (AAA) switch functionalities on any switch in the Cisco MDS 9000 Family. This section includes the following topics:

Authentication

Authentication is the process of verifying the identity of the person managing the switch. This identity verification is based on the user ID and password combination provided by the person trying to manage the switch. Cisco MDS 9000 Family switches allow you to perform local authentication (using the lookup database) or remote authentication (using one or more RADIUS or TACACS+ servers).

Authorization

By default, two roles exist in all switches:

- Network operator (`network-operator`)—Has permission to view the configuration only. The operator cannot make any configuration changes.
- Network administrator (`network-admin`)— Has permission to execute all commands and make configuration changes. The administrator can also create and customize up to 64 additional roles.

The two default roles cannot be changed or deleted. You can create additional roles and configure the following options:

- Configure role-based CLI authorization by assigning user roles locally or using remote AAA servers
- Configure CLI user profiles on a remote AAA server to contain role information. This role information is automatically downloaded and used when the user is authenticated through the remote AAA server

Accounting

Accounting refers to the log that is kept for each management session in a switch. This information may be used to generate reports for troubleshooting purposes and user accountability. Accounting can be implemented locally and remotely.

Remote Authentication by AAA Servers

AAA authentication provides the following advantages over local database authentication:

- It requires only one password to be shared between the switch and the AAA servers.
- It is easier to manage user password lists for each switch in the fabric.
- AAA servers are deployed widely across enterprises and can be easily adopted.

Remote Authentication Guidelines

When you prefer using remote C servers, follow these guidelines:

- A minimum of one AAA server should be IP reachable.
- If all configured AAA servers are not reachable, the policy configured on the switch determines the authentication method.
- RADIUS servers are easily reachable if an overlay Ethernet LAN is attached to the switch (see Chapter 22, “Configuring IP Storage”). This is the recommended method.
- SAN networks connected to the switch should have at least one gateway switch connected to the Ethernet LAN containing the AAA servers

Note If you are using IP connectivity to reach an AAA server, the SAN connects to the switch.

Server Groups

You can specify remote AAA servers for authentication, authorization, and accounting using server groups. A server group consists of remote AAA servers implementing the same AAA protocol. The purpose of a server group is to provide for fail-over servers in case a remote AAA server fails to respond. If the first remote server in the group fails to respond, the next remote server in the group is tried until one of the servers sends a response. If all the AAA servers in the server group fails to respond, then that server group option is considered a failure. You can create a server group using the **aaa group server** command. If required, you can specify multiple server groups. If the MDS switch encounters errors from the server(s) in the first group, it tries the servers in the next server group.

AAA Service Configuration Options

AAA configuration in Cisco MDS 9000 Family switches is service based. You can have separate AAA configurations for the following services

- Telnet or SSH login—Use the **aaa authentication login default** command.
- Console login—Use the **aaa authentication login console** command.
- iSCSI authentication—Use the **aaa authentication iscsi default** command (see the “Authentication Mechanism” section on page 22-65).
- FC-SP authentication—Use the **aaa authentication dhchap default** command (see Chapter 17, “Configuring Fabric Security”).
- Accounting—use the **aaa accounting default** command

In general, server group, local, and none are the three options that can be specified for any service in an AAA configuration. Each option is tried in the order specified. If all the methods fail, local is tried.

Caution Cisco MDS SAN-OS does not support all numeric usernames, whether created with TACACS+ or RADIUS, or created locally. Local users with all numeric names cannot be created. If an all numeric username exists on an AAA server and is entered during login, the user is not logged in.

Note Even if local is not specified as one of the options, it is tried when all other configured options fail.

Configuring RADIUS

Cisco MDS 9000 Family switches use the RADIUS protocol to communicate with remote AAA servers. You can configure multiple RADIUS servers and set timeout and retry counts.

This section defines the RADIUS operation, identifies its network environments, and describes its configuration possibilities. This section includes the following topics:

About RADIUS

RADIUS is a distributed client/server system that secures networks against unauthorized access. In the Cisco implementation, RADIUS clients run on Cisco MDS 9000 Family switches and send authentication requests to a central RADIUS server that contains all user authentication and network service access information.

RADIUS is a fully open protocol, distributed in source code format, that can be modified to work with any security system currently available on the market.

You can set the RADIUS server address, the RADIUS preshared key, the RADIUS server timeout interval, iterations of the RADIUS server, define vendor-specific attributes, and display RADIUS server details.

Setting the RADIUS Server Address

You can add up to 64 RADIUS servers using the **radius-server host** command. RADIUS keys are always stored in encrypted form in persistent storage. The running configuration also displays encrypted keys.

To specify the RADIUS server address and the options, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# radius-server host 10.10.0.0 key HostKey`	Specifies a key for the selected RADIUS server. This key overrides the key assigned using the **radius-server key** command. In this example, the host is 10.10.0.0 and the key is HostKey.
Step 3	`switch(config)# radius-server host 10.10.0.0 auth-port 2003`	Specifies the destination UDP port number to which the RADIUS authentication messages should be sent. In this example, the host is 10.10.0.0 and the authentication port is 2003. The default authentication port is 1812, and the valid range is 0 to 65366.
Step 4	`switch(config)# radius-server host 10.10.0.0 acct-port 2004`	Specifies the destination UDP port number to which RADIUS accounting messages should be sent. The default accounting port is 1813, and the valid range is 0 to 65366.
Step 5	`switch(config)# radius-server host 10.10.0.0 accounting`	Specifies this server to be used only for accounting purposes. **Note** If neither the **authentication** nor the **accounting** options are specified, the server is used for both accounting and authentication purposes.
Step 6	`switch(config)# radius-server host radius2 key 0 abcd`	Specifies a clear text key for the specified server. The key is restricted to 65 characters.
	`switch(config)# radius-server host radius3 key 7 1234`	Specifies a reversible encrypted key for the specified server. The key is restricted to 65 characters.

Setting the RADIUS Preshared Key

You need to configure the RADIUS preshared key to authenticate the switch to the RADIUS server. The length of the key is restricted to 65 characters and can include any printable ASCII characters (white spaces are not allowed). You can configure a global key to be used for all RADIUS server configurations on the switch. You can override this global key assignment by explicitly using the **key** option in the **radius-server host** command.

To set the RADIUS preshared key, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.

	Command	Purpose
Step 2	`switch(config)# radius-server key AnyWord`	Configures a preshared key (AnyWord) to authenticate communication between the RADIUS client and server. The default is clear text.
	`switch(config)# radius-server key 0 AnyWord`	Configures a preshared key (AnyWord) specified in clear text (indicated by 0) to authenticate communication between the RADIUS client and server.
	`switch(config)# radius-server key 7 public`	Configures a preshared key (public) specified in encrypted text (indicated by 7) to authenticate communication between the RADIUS client and server.

Setting the RADIUS Server Time-Out Interval

To specify the time between retransmissions to the RADIUS servers, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# radius-server timeout 30`	Specifies the time (in seconds) between retransmissions to the RADIUS server. The default timeout is one (1) second. The time range in seconds is 1 to 60.

You can revert the retransmission time to its default by issuing the **no radius-server timeout** command.

Setting Iterations of the RADIUS Server

By default, a switch retries a RADIUS server connection only once. This number can be configured. The maximum is five retries per server. You can revert the retry number to its default by issuing the **no radius-server retransmit** command.

To specify the number of times that RADIUS servers should try to authenticate a user, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# radius-server retransmit 3`	Configures the number of times (3) the switch tries to connect to a RADIUS server(s) before reverting to local authentication.

Defining Vendor-Specific Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific attributes (VSAs) between the network access server and the RADIUS server. The IETF uses attribute 26. VSAs allow vendors to support their own extended attributes that are not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option using the format recommended in the specification. The Cisco vendor ID is 9, and the supported option is vendor type 1, which is named `cisco-avpair`. The value is a string with the following format:

```
protocol : attribute sep value *
```

Where `protocol` is a Cisco attribute for a particular type of authorization, and `sep` is `=` for mandatory attributes, and `*` is for optional attributes.

When you use RADIUS servers to authenticate yourself to a Cisco MDS 9000 Family switch, the RADIUS protocol directs the RADIUS server to return user attributes, like authorization information, along with authentication results. This authorization information is specified through VSAs.

VSA Format

The following VSA protocol options are supported:

- Shell protocol—used in access-accept packets to provide user profile information.
- Accounting protocol—used in accounting-request packets. If a value contains any white spaces, it should be put within double quotation marks.

The following attributes are supported:

- `roles`–This attribute lists all the roles to which the user belongs. The value field is a string storing the list of group names delimited by white space. For example, if you belong to roles `vsan-admin` and `storage-admin`, the value field would be "`vsan-admin storage-admin`." This subattribute is sent in the VSA portion of the Access-Accept frames from the RADIUS server, and it can only be used with the shell protocol value. This is an example using the roles attribute:

  ```
  Cisco-AVPair = shell:roles="network-admin vsan-admin"
  ```

- `accountinginfo`—This attribute stores additional accounting information besides the attributes covered by a standard RADIUS accounting protocol. This attribute is only sent in the VSA portion of the Account-Request frames from the RADIUS client on the switch, and it can only be used with the accounting protocol value.

Displaying RADIUS Server Details

Use the **show radius-server** command to display configured RADIUS parameters (see Example 16-1).

Note Only administrators can view the RADIUS preshared key.

Example 16-1 Displays Configured RADIUS Information

```
switch# show radius-server
Global RADIUS shared secret:Myxgqc
retransmission count:5
timeout value:10
following RADIUS servers are configured:
        myradius.cisco.users.com:
```

```
        available for authentication on port:1812
        available for accounting on port:1813
    172.22.91.37:
        available for authentication on port:1812
        available for accounting on port:1813
        RADIUS shared secret:23MHcUnD
    10.10.0.0:
        available for authentication on port:1812
        available for accounting on port:1813
        RADIUS shared secret:hostkey----> for administrators only
```

Example 16-2 Displays Configured RADIUS Server-Group Order

```
switch# show radius-server groups
total number of groups:4
following RADIUS server groups are configured:
        group radius:
                server: all configured radius servers
        group Group1:
                server: Server3 on auth-port 1812, acct-port 1813
                server: Server5 on auth-port 1812, acct-port 1813
        group Group5:
```

Configuring TACACS+

A Cisco MDS switch uses the Terminal Access Controller Access Control System Plus (TACACS+) protocol to communicate with remote AAA servers. You can configure multiple TACACS+ servers and set timeout values. This section includes the following topics:

About TACACS+

TACACS+ is a client/server protocol that uses TCP (TCP port 49) for transport requirements. All switches in the Cisco MDS 9000 Family provide centralized authentication using the TACACS+ protocol. The addition of TACACS+ support in Cisco SAN-OS 1.3 enables the following advantages over RADIUS authentication:

- Provides independent, modular AAA facilities. Authorization can be done without authentication.
- Performs independent of servers if it is configured to its own database.
- TCP transport protocol to send data between the AAA client and server, using reliable transfers with a connection-oriented protocol.
- Encrypts the entire protocol payload between the switch and the AAA server to ensure higher data confidentiality. The RADIUS protocol only encrypts passwords.

Advantages of TACACS+

This section provides a brief list of advantages that TACACS+ has over and RADIUS.

- Uses a TCP protocol that has a connection-oriented transport
- Provides built-in transport support
- Provides a separate acknowledgment that a request has been received
- Provides immediate indication of a crashed, or failed, server
- Detects server crashes out-of-band with actual requests
- Maintains simultaneous connections to multiple servers
- Adapts to growing and congested networks

For a detailed comparison, visit the following URL:

http://www.cisco.com/warp/public/480/10.html#comparing

Enabling TACACS+

By default, the TACACS+ feature is disabled in all switches in the Cisco MDS 9000 Family. You must explicitly enable the TACACS+ feature to access the configuration and verification commands for fabric authentication. When you disable this feature, all related configurations are automatically discarded.

To enable TACACS+ for a Cisco MDS switch, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# tacacs+ enable`	Enables the TACACS+ in this switch.
	`switch(config)# no tacacs+ enable`	Disables (default) the TACACS+ in this switch.

Setting the TACACS+ Server Address

Use the **tacacs-server** command to configure the communication parameters for the required TACACS+ server.

If a secret key is not configured for a configured server, a warning message is issued and the global secret encryption key is automatically used (see the "Setting the Secret Key" section on page 16-12).

To configure the TACACS+ server option, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# tacacs-server host 171.71.58.91` `warning: no key is configured for the host`	Configures the TACACS+ server identified by the specified IP address.
	`switch(config)# no tacacs-server host 10.10.1.0`	Deletes the specified TACACS+ server identified by the IP address. By default, no server is configured.
Step 3	`switch(config)# tacacs-server host 171.71.58.91 port 2`	Configures the TCP port for all TACACS+ requests.
	`switch(config)# no tacacs-server host 171.71.58.91 port 2`	Reverts to the factory default of using Port 49 for server access.
Step 4	`switch(config)# tacacs-server host host1.cisco.com key MyKey`	Configures the TACACS+ server identified by the specified domain name and assigns a special key.
Step 5	`switch(config)# tacacs-server host host100.cisco.com timeout 25`	Configures the timeout period for the switch to wait for a response from the specified server before it declares a timeout failure.

Setting the Secret Key

Use the **tacacs-server** command to configure global values for the **key** for all TACACS+ servers.

Note Secret keys configured for individual servers override the globally configured values.

To set the secret key for TACACS+ servers, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# tacacs-server key 7 tacacsPword`	Assigns the global secret key to access the TACACS+ server. This example specifies **7** to indicate encryption. If this global key and the individual server keys are not configured, clear text messages are sent to the TACACS+ server(s). **Note** If secret keys are configured for individual servers, those keys override this global key.
	`switch(config)# no tacacs-server key oldPword`	Deletes the configured secret key to access the TACACS+ server and reverts to the factory default of allowing access to all configured servers.

Setting the Timeout Value

Use the **tacacs-server** command to configure global **timeout** values for all TACACS+ servers.

Note Timeout values configured for individual servers override the globally configured values.

To set the share password for TACACS+ servers, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# tacacs-server timeout 30`	Configures the global timeout period for the switch to wait for a response from all servers before it declares a timeout failure.
	`switch(config)# no tacacs-server timeout 30`	Deletes the configured timeout period and reverts to the factory default of 5 seconds. **Note** If the timeout value is configured for individual servers, that value overrides this global timeout value.

Defining Custom Attributes for Roles

Cisco MDS 9000 Family switches use the TACACS+ custom attribute for service shells to configure roles to which a user belongs. TACACS+ attributes are specified in `name=value` format. The attribute name for this custom attribute is cisco-av-pair. The following example illustrates how to specify roles using this attribute:

```
cisco-av-pair=shell:roles="network-admin vsan-admin"
```

Note TACACS+ custom attributes can be defined on an ACS server for various services (for example, shell). Cisco MDS 9000 Family switches require the TACACS+ custom attribute for the service shell to be used for defining roles.

Displaying TACACS+ Server Details

Use the **show tacacs+** commands to display configurations for the TACACS+ protocol configuration in all switches in the Cisco MDS 9000 Family (see Examples 16-3 to 16-6).

Example 16-3 Displays Configured TACACS+ Server Information

```
switch# show tacacs-server
Global TACACS+ shared secret:tacacsPword
timeout value:30
total number of servers:3

following TACACS+ servers are configured:
        171.71.58.91:
                available on port:2
        cisco.com:
                available on port:49
        171.71.22.95:
                available on port:49
                TACACS+ shared secret:MyKey
```

Example 16-4 Displays AAA Authentication Information

```
switch# show aaa authentication
         default: group TacServer local none
         console: local
         iscsi: local
         dhchap: local
```

Example 16-5 Displays Configured TACACS Server Groups

```
switch# show tacacs-server groups
total number of groups:2

following TACACS+ server groups are configured:
        group TacServer:
                server 171.71.58.91 on port 2
        group TacacsServer1:
                server ServerA on port 49
                server ServerB on port 49:
```

Example 16-6 Displays All AAA Server Groups

```
switch# show aaa groups
radius
TacServer
```

Configuring Server Groups

You can specify one or more remote AAA servers to authenticate users using server groups. All members of a group must belong to the same protocol: either RADIUS or TACACS+. The servers are tried in the same order in which you configure them.

You can configure these server groups at any time but they only take effect when you apply them to a AAA service using the **aaa authentication login** command (see the "Defining Custom Attributes for Roles" section on page 16-13).

You can specify one or more remote AAA servers to authenticate users using server groups.

To specify the TACACS+ server order within a group, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# aaa group server tacacs+ TacacsServer1` `switch(config-tacacs+)#`	Configures a TacacsServer1 group and enters the submode for that group.
	`switch(config)# no aaa group server tacacs+ TacacsServer19`	Deletes the group called TacacsServer19 from the authentication list.
Step 3	`switch(config-tacacs+)# server ServerA`	Configures ServerA to be tried first within TacacsServer1. **Tip** If the specified TACACS+ server is not found, configure it using the **tacacs-server host** command and retry this command.
Step 4	`switch(config-tacacs+)# server ServerB`	Configures ServerB to be tried second within TacacsServer1.
	`switch(config-tacacs+)# no server ServerZ`	Deletes ServerZ within the TacacsServer1 list of servers.

To verify the configured server-group order, use the **show tacacs-server groups** command:

```
switch# show tacacs-server groups
total number of groups:2

following TACACS+ server groups are configured:
        group TacServer:
                server 171.71.58.91 on port 2
        group TacacsServer1:
                server ServerA on port 49
                server ServerB on port 49:
```

Local AAA

The system maintains the user name and password locally and stores the password information in encrypted form. You are authenticated based on the locally stored user information. You can configure local users using the **username** command (see the "Creating or Updating Users" section on page 16-22). You can view the local accounting log using the **show accounting log** command (see Example 16-7).

Example 16-7 Displays the Accounting Log Information

```
switch# show accounting log

Sat Jan 24 03:22:06 1981:stop:snmp_349154526_171.71.58.69:admin:
Sat Jan 24 03:22:06 1981:start:snmp_349154526_171.71.58.69:admin:
Sat Jan 24 03:22:06 1981:update:snmp_349154526_171.71.58.69:admin:Added member [
 WWN: 21:00:00:20:37:a6:be:00 ID: 2] to zone test-27 on VSAN 1
...
Sat Jan 24 23:59:56 1981:stop:/dev/pts/0_349228792:root:shell terminated
Sun Jan 25 00:00:06 1981:start:/dev/pts/1_349228806:admin:
```

No AAA Authentication

You can turn off password verification using the **none** option in the **aaa authentication login** command. If you configure this option, users can login without giving a valid password. But the user should at least exist locally on the Cisco MDS 9000 Family switch.(created using the **username** command).

Caution Use this option cautiously. If configured, any user will be able to access the switch at any time.

Displaying AAA Authentication

The **show aaa authentication** command displays the configured authentication methods (see Example 16-8).

Example 16-8 Example 16-8 Displays Authentication Information

```
switch# show aaa authentication

    No AAA Authentication
    default: group TacServer local none
    console: local none
    iscsi: local
    dhchap: local
```

Authentication and Authorization Process

Authentication is the process of verifying the identity of the person managing the switch. This identity verification is based on the user ID and password combination provided by the person trying to manage the switch. The Cisco MDS 9000 Family switches allow you to perform local authentication (using the lookup database) or remote authentication (using one or more RADIUS servers or TACACS+ servers).

Figure 16-1 shows a flow chart of the process.

The following steps explain the authorization and authentication process.

Step 1 When you can log in to the required switch in the Cisco MDS 9000 Family, you can use the Telnet, SSH, or console login options.

- For Telnet/SSH log in, use the **aaa authentication login default** command.
- For console log in, use the **aaa authentication login console** command. If this command is not configured, the software automatically uses the **aaa authentication login default** command.

Step 2 When you configure server groups using the server group authentication method, an authentication request is sent to the first AAA server in the group.

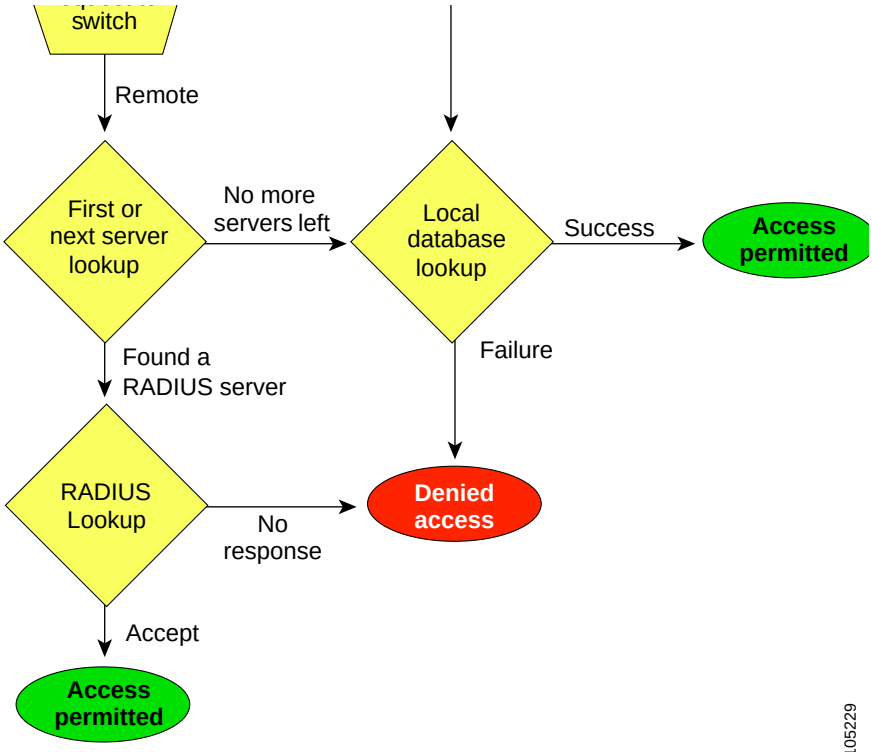
- If the AAA server fails to respond, then the next AAA server is tried and so on until the remote server responds to the authentication request.
- If all AAA servers in the server group fail to respond, then the servers in the next server group are tried.
- If all configured methods fail, then the local database is used for authentication.

Step 3 If you are successfully authenticated through a remote AAA server, then the following possibilities apply.

- If AAA server protocol is RADIUS, then user roles specified in the cisco-av-pair attribute are downloaded with an authentication response.
- If AAA server protocol is TACACS+, then another request is sent to the same server to get the user roles specified as custom attributes for the shell.
- If user roles are not successfully retrieved from the remote AAA server, then the user is assigned the network-operator role.

Step 4 If your user name and password are successfully authenticated, you are allowed to log in.

Figure 16-1 Switch Authorization and Authentication Flow[1] [2]



1. No more server groups left = no response from any server in all server groups.
2. No more servers left = no response from any server within this server group.

Role-Based CLI Authorization

Switches in the Cisco MDS 9000 Family perform authentication based on roles. Role-based authorization limits access to switch operations by assigning users to roles. This kind of authentication restricts you to management operations based on the roles to which you have been added.

When you execute a command, perform command completion, or obtain context sensitive help, the switch software allows the operation to progress if you have permission to access that command.

Each role can contain multiple users and each user can be part of multiple roles. For example, if role1 users are only allowed access to configuration commands, and role2 users are only allowed access to debug commands, then if Joe belongs to both role1 and role2, he can access configuration as well as debug commands.

Note If you belong to multiple roles, you can execute a superset of all the commands permitted by these roles. Access to a command takes priority over being denied access to a command. For example, suppose you belong to a TechDocs group and you were denied access to configuration commands. However, you also belong to the engineering group and have access to configuration commands. In this case, you will have access to configuration commands.

Tip Any role, when created, does not allow access to the required commands immediately. The administrator must configure appropriate rules for each role to allow access to the required commands.

Configuring Roles and Profiles

To create an additional role or to modify the profile for an existing role, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# role name techdocs` `switch(config-role)#`	Places you in the mode for the specified role (techdocs). **Note** The role submode prompt indicates that you are now in the role submode. This submode is now specific to the techdocs group.
	`switch(config)# no role name techdocs`	Deletes the role called techdocs.
Step 3	`switch(config-role)# description Entire Tech. Docs. group`	Assigns a description to the new role. The description is limited to one line and can contain spaces.
	`switch(config-role)# no description`	Resets the description for the Tech. Docs. group.

Configuring Rules and Features for Each Role

Tip A user not belonging to the network-admin role cannot perform commands related to roles. For example, if user A is permitted to perform all **show** commands, user A cannot view the output of the **show role** command if user A does not belong to the network-admin role

The **rule** command specifies operations that can be performed by a specific role. Each rule consists of a rule number, a rule type (permit or deny), a command type (for example, **config**, **clear**, **show**, **exec**, **debug**), and an optional feature name (for example, FSPF, zone, VSAN, fcping, or interface).

The user-specified rule number determines the order in which the rules are applied. For example, rule 1 is applied before rule 2, which is applied before rule 3, and so on.

Note In this case, **exec** commands refer to all commands in the EXEC mode that do not fall in the **show**, **debug**, and **clear**, categories. Up to 16 rules can be configured for each role.

Modifying Profiles

To modify the profile for an existing role, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# role name sangroup` `switch(config-role)#`	Places you in sangroup role submode.

Step 3	`switch(config-role)# rule 1 permit config`
	`switch(config-role)# rule 2 deny config feature fspf`
	`switch(config-role)# rule 3 permit debug feature zone`
	`switch(config-role)# rule 4 permit exec feature fcping`
		Allows users belonging to the sangroup role to perform all configuration commands except f**spf config** commands. They can also perform **zone debug** commands and the **fcping** EXEC mode command.

Step 4	`switch(config-role)# no rule 4`	Deletes rule 4, which no longer permits the sangroup to perform the **fcping** command.

In Step 3, rule 1 is applied first, thus permitting sangroup users access to all **config** commands. Rule 2 is applied next, denying FSPF configuration to sangroup users. As a result, sangroup users can perform all other **config** commands, except **fspf** configuration commands.

Note The order of rule placement is important. If you had swapped these two rules and issued the **deny config feature fspf** rule first and issued the **permit config** rule next, you would be allowing all sangroup users to perform all configuration commands because the second rule globally overrode the first rule.

Configuring the VSAN Policy

Configuring the VSAN policy requires the ENTERPRISE_PKG license (see "Obtaining and Installing Licenses" section on page 3-1).

You can configure a role so that it only allows commands to be performed for a selected set of VSANs. By default, the VSAN policy for any role is **permit**. In other words, the role can perform commands configured by the **rule** command in all VSANs. To selectively allow VSANs for a role, the VSAN policy needs to be set to **deny**, and then the appropriate VSANs need to be permitted.

Note Users configured in roles where the VSAN policy is set to **deny** cannot modify the configuration for E ports. They can only modify the configuration for F or FL ports (depending on whether the configured rules allow such configuration to be made). This is to prevent such users from modifying configurations that may impact the core topology of the fabric.

Tip Roles can be used to create VSAN administrators. Depending on the configured rules, these VSAN administrators can configure MDS features (for example, zone, fcdomain, or VSAN properties) for their VSANs without affecting other VSANs. Also, if the role permits operations in multiple VSANs, then the VSAN administrators can change VSAN membership of F or FL ports among these VSANs.

Users belonging to roles in which the VSAN policy is set to **deny** are referred to as VSAN-restricted users. These users cannot perform commands that require the startup configuration to be viewed or modified. These commands include the **copy running startup**, **show startup**, **show running-config diff**, **copy startup** <*destination*>, and **copy** <*source*> **startup** commands. For information on these commands, see Chapter 2, "Before You Begin."

Modifying the VSAN Policy

To modify the VSAN policy for an existing role, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# role name sangroup` `switch(config-role)#`	Places you in sangroup role submode.
Step 3	`switch(config)# vsan policy deny` `switch(config-role-vsan)`	Changes the VSAN policy of this role to **deny** and places you in a submode where VSANs can be selectively permitted.
	`switch(config-role)# no vsan policy deny`	Deletes the configured VSAN role policy and reverts to the factory default (**permit**).
Step 4	`switch(config-role-vsan)# permit vsan 10-30`	Permits this role to perform the allowed commands for VSANs 10 through 30.
	`switch(config-role-vsan)# no permit vsan 15-20`	Removes the permission for this role to perform commands for vsan 15 to 20. So, the role is now permitted to perform commands for VSAN 10 to 14, and 21 to 30.

Displaying Role-Based CLI Information

Use the **show role** command to display rules configured on the switch including those rules that have not yet been committed to persistent storage. The rules are displayed by rule number and are based on each role. All roles are displayed even if the role name is not specified. See Example 16-9.

Example 16-9 Displays Information for All Roles

```
switch# show role
Role: network-admin
Description: Predefined Network Admin group. This role cannot be modified
Access to all the switch commands

Role: network-operator
Description: Predefined Network Operator group. This role cannot be modified
Access to Show commands and selected Exec commands

Role: TechDocs
vsan policy: permit (default)

Role: sangroup
Description: SAN management group
vsan policy: deny
Permitted vsans: 10-30

----------------------------------------
Rule   Type   Command-type   Feature
----------------------------------------
  1.   permit    config           *
  2.     deny    config        fspf
  3.   permit     debug        zone
  4.   permit      exec      fcping
```

Configuring CLI User Profiles

Every Cisco MDS 9000 Family switch user has related NMS information stored by the system. Your authentication information, user name, user password, password expiration date, and role membership are stored in your user profile. The CLI commands explained in this section enable you to create users and modify the profile of an existing user.These commands are restricted to privileged users as determined by your administrator.

Creating or Updating Users

Cisco MDS 9000 Family switches use the same command (**username**) to create a user and to update an existing user. The **expire** option determines the date on which the user account is disabled. The date is specified in the YYYY-MM-DD format. By default, the user account does not expire unless you explicitly configure it to expire.

Tip The following words are reserved and cannot be used to configure users: bin, daemon, adm, lp, sync, shutdown, halt, mail, news, uucp, operator, games, gopher, ftp, nobody, nscd, mailnull, rpc, rpcuser, xfs, gdm, mtsuser, ftpuser, man, and sys.

Note User passwords are not displayed in the switch configuration file.

To configure a new user or to modify the profile of an existing user, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# username usam password abcd expire 2003-05-31`	Creates or updates the user account (usam) along with a password (abcd) that is set to expire on 2003-05-31. The password is limited to 64 characters.
	`switch(config)# username msam password 0 abcd role network-operator`	Creates or updates the user account (msam) along with a password (abcd) specified in clear text (indicated by 0). The password is limited to 64 characters.
	`switch(config)# username user1 password 5 !@*asdsfsdfjh!@df`	Specifies an encrypted (specified by 5) password (!@*asdsfsdfjh!@df) for the user account (user1).
Step 3	`switch(config)# username usam role network-admin`	Adds the specified user (usam) to the network-admin role.
	`switch(config)# no username usam role vsan-admin`	Deletes the specified user (usam) from the vsan-admin role.
Step 4	`switch(config)# username usam sshkey fsafsd2344234234ffgsdfg`	Specifies the SSH key for the user account (usam).
	`switch(config)# no username usam sshkey fsafsd2344234234ffgsdfgffsdfsfsfssf`	Deletes the SSH key for the user account (usam).

Note If the **update-snmpv3** option is used, specify the clear text and old SNMP password (see the "Forcing Identical SNMP and CLI Passwords" section on page 16-34).

Logging out CLI Users

To log out another user on the switch, use the **clear user** command. In the following example, the user named vsam is logged out from the switch.

```
switch# clear user vsam
```

Use the **show users** command to view a list of the logged in users (see Example 16-10).

Example 16-10 Displays All Logged in Users

```
switch# show users
admin    pts/7        Jan 12 20:56 (10.77.202.149)
admin    pts/9        Jan 12 23:29 (modena.cisco.com)
admin    pts/10       Jan 13 03:05 (dhcp-171-71-58-120.cisco.com)
admin    pts/11       Jan 13 01:53 (dhcp-171-71-49-49.cisco.com)
```

Displaying User Profile Information

Use the **show user-account** command to display configured information about user accounts. See Examples 16-11 to 16-12.

Example 16-11 Displays Information for a Specified User

```
switch# show user-account user1
user:user1
        this user account has no expiry date
        roles:network-operator
no password set. Local login not allowed
Remote login through RADIUS is possible
```

Example 16-12 Displays Information for All Users

```
switch# show user-account
show user-account
user:admin
        this user account has no expiry date
        roles:network-admin
user:usam
        expires on Sat May 31 00:00:00 2003
        roles:network-admin network-operator
user:msam
        this user account has no expiry date
        roles:network-operator
user:user1
        this user account has no expiry date
        roles:network-operator
no password set. local login not allowed
Remote login through RADIUS is possible
```

Configuring CLI Accounting Parameters

Accounting refers to the log information that is kept for each management session in a switch. This information may be used to generate reports for troubleshooting purposes and user accountability. Accounting can be implemented locally or remotely (using RADIUS).

Setting the Accounting Log Size

The **aaa accounting logsize** command sets the size limit of the accounting log file in persistent storage. The default is 15,000 bytes.

To set the log file size, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# aaa accounting logsize 29000`	Sets the size of the log file on the local disk. The default is 15,000 bytes.

Tip The Cisco MDS 9000 Family switch uses Interim-Update RADIUS accounting-request packets to communicate accounting log information to the RADIUS server. The RADIUS server must be appropriately configured to log the information communicated in these packets. Several servers typically have Log Update/Watchdog Packets flag in the AAA client configuration. This flag should be turned on to ensure proper RADIUS accounting.

Note Configuration operations are automatically recorded in the accounting log if they are performed in configuration mode. Additionally, important system events (for example, configuration save and system switchover) are also recorded in the accounting log.

Displaying Accounting Configuration

The **show accounting** command displays configured accounting information. See Examples 16-13 to 16-15.

Example 16-13 Displays Configured Accounting Parameters.

```
switch# show accounting config
RADIUS accounting not enabled
local accounting enabled
```

Example 16-14 Displays Configured Log Size.

```
switch# show accounting logsize
maximum local accounting log size:29000
```

Example 16-15 Displays the Entire Log File.

```
switch# show accounting log
```

```
Fri Jan 16 15:28:21 1981:stop:snmp_348506901_64.104.131.208:admin:
Fri Jan 16 21:17:04 1981:start:/dev/pts/0_348527824:admin:
Fri Jan 16 21:35:45 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
group:Group1
Fri Jan 16 21:35:51 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
group:Group1
Fri Jan 16 21:35:51 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
group:Group5
Fri Jan 16 21:35:55 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
group:Group5
Fri Jan 16 21:35:55 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
group:Group3
Fri Jan 16 21:58:17 1981:start:snmp_348530297_171.71.150.105:admin:
Fri Jan 16 21:58:17 1981:stop:snmp_348530297_171.71.150.105:admin:
Fri Jan 16 21:58:18 1981:start:snmp_348530298_171.71.150.105:admin:
Fri Jan 16 21:58:18 1981:stop:snmp_348530298_171.71.150.105:admin:
...
Fri Jan 16 23:37:02 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
group:Group3
Fri Jan 16 23:37:26 1981:update:/dev/pts/0_348527824:admin:updated TACACS+ parameters for
group:TacacsServer1
Fri Jan 16 23:45:19 1981:update:/dev/pts/0_348527824:admin:updated TACACS+ parameters for
group:TacacsServer1
Fri Jan 16 23:45:19 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
group:Group1
...
Fri Jan 16 23:53:51 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
server:Server3
Fri Jan 16 23:54:00 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
server:Server5
Fri Jan 16 23:54:22 1981:update:/dev/pts/0_348527824:admin:updated TACACS+ parameters for
server:ServerA
Fri Jan 16 23:54:25 1981:update:/dev/pts/0_348527824:admin:updated TACACS+ parameters for
server:ServerB
Fri Jan 16 23:55:03 1981:update:/dev/pts/0_348527824:admin:updated RADIUS parameters for
group:Group1
...
Sat Jan 17 00:01:41 1981:start:snmp_348537701_171.71.58.100:admin:
Sat Jan 17 00:01:41 1981:stop:snmp_348537701_171.71.58.100:admin:
Sat Jan 17 00:01:42 1981:start:snmp_348537702_171.71.58.100:admin:
Sat Jan 17 00:01:42 1981:stop:snmp_348537702_171.71.58.100:admin:
...
```

Recovering Administrator Password

An administrator can recover a password from a local console connection. The password recovery procedure must be performed on the supervisor module that becomes the active supervisor module after the recovery procedure is completed. To ensure the other supervisor module does not become the active module, you have two options:

- Physically remove the other supervisor module from the chassis, or
- For the duration of this procedure, change the other supervisor module's console prompt to the `loader>` or `switch(boot)#` prompt (see Chapter 6, “Software Images”).

Note Password recovery is not possible from a Telnet or SSH session.

To recover a administrator’s password, follow these steps:

Step 1 Reboot the switch.

```
switch# reload
The supervisor is going down for reboot NOW!
```

Step 2 Press the **Ctrl-]** key sequence (when the switch begins its Cisco SAN-OS software boot sequence) to enter the `switch(boot)#` prompt (see the “Recovery Interruption” section on page 6-24).

```
Ctrl-]
switch(boot)#
```

Step 3 Change to configuration mode.

```
switch(boot)# config terminal
```

Step 4 Enter the **admin-password** command to reset the administrator password.

```
switch(boot-config)# admin-password password
```

Step 5 Exit to the EXEC mode.

```
switch(boot-config)# exit
switchboot#
```

Step 6 Enter the **load** command to load the Cisco SAN-OS software.

```
switch(boot)# load bootflash:system.img
```

Step 7 Save the software configuration.

```
switch# copy running-config startup-config
```

Configuring SSH Services

The Telnet service is enabled by default on all Cisco MDS 9000 Family switches. Before enabling the SSH service, generate a host key pair. To generate a host key, use the **ssh key** command (see the "Generating the SSH Host Key Pair" section on page 16-28).

Enabling SSH Service

By default, the SSH service is disabled. To enable SSH service, issue the **ssh server enable** command.

To enable or disable the SSH service, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# ssh server enable updated`	Enables the use of the SSH service.
	`switch(config)# no ssh server enable updated`	Disables (default) the use of the SSH service and resets the switch to its factory defaults.

Caution If you are logging in to a switch through SSH and you have issued the **aaa authentication login default none** command, you must enter one or more key strokes to log in. If you press the **Enter** key without entering at least one keystroke, your log in will be rejected.

Specifying the SSH Key

You can specify a SSH key to log in using the SSH client without being prompted for a password.

To specify or delete the SSH Key for a specified user, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# username usam sshkey fsafsd2344234234ffgsdfg`	Specifies the SSH key for the user account (usam).
	`switch(config)# no username usam sshkey fsafsd2344234234ffgsdfgffsdfsfsfssf`	Deletes the SSH key for the user account (usam).

Generating the SSH Host Key Pair

Be sure to have an SSH host key pair with the appropriate version before enabling the SSH service. Generate the SSH host key pair according to the SSH client version used. The number of bits specified for each key pair ranges from 768 to 2048.

The SSH service accepts three types of key pairs for use by SSH versions 1 and 2.

- The **rsa1** CLI option generates the RSA1 key pair for the SSH version 1 protocol.
- The **dsa** CLI option generates the DSA key pair for the SSH version 2 protocol.
- The **rsa** CLI option generates the RSA key pair for the SSH version 2 protocol.

To generate the SSH host key pair, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.

Step 2	`switch(config)# ssh key rsa1 1024`
	`generating rsa1 key.....`
	`generated rsa1 key`
		Generates the RSA1 host key pair.

	`switch(config)# ssh key dsa 1024`
	`generating dsa key.....`
	`generated dsa key`
		Generates the DSA host key pair.

	`switch(config)# ssh key rsa 1024`
	`generating rsa key.....`
	`generated rsa key`
		Generates the RSA host key pair.

	`switch(config)# no ssh key rsa 1024` `cleared RSA keys`	Clears the RSA host key pair configuration. **Caution** If you delete all of the SSH keys, you cannot start a new session.

Overwriting a Generated Key Pair

If the SSH key pair option is already generated for the required version, use the **force** option to overwrite the previously generated key pair.

To overwrite the previously generated key pair, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.

Step 2	`switch(config)# ssh key dsa 768`
	`ssh key dsa 512`
	`dsa keys already present, use force`
	`option to overwrite them`
	`switch(config)# ssh key dsa 512 force`
	`deleting old dsa key.....`
	`generating dsa key.....`
	`generated dsa key`
	`switch(config)#`
		Tries to set the host key pair. If a required host key pair is already configured, use the **force** option to overwrite that host key pair.
		Deletes the old DSA key and sets the host key pair using the new bit specification.

Clearing SSH Hosts

Use the **clear ssh hosts** command to manually clear trusted SSH host entries (see Example 16-6):

Example 16-16 Clearing Configured SSH Hosts

```
switch# clear ssh hosts
```

This command clears reset-reason information from NVRAM and volatile storage.

Displaying SSH Protocol Status

Use the **show ssh server** command to display the status of the SSH protocol (enabled or disabled) and the versions that are enabled for that switch (see Example 16-17).

Example 16-17 Displays SSH Protocol Status

```
switch# show ssh server
ssh is enabled
version 1 enabled
version 2 enabled
```

Use the **show ssh key** command to display the host key pair details for the specified key or for all keys, if no key is specified (see Example 16-18).

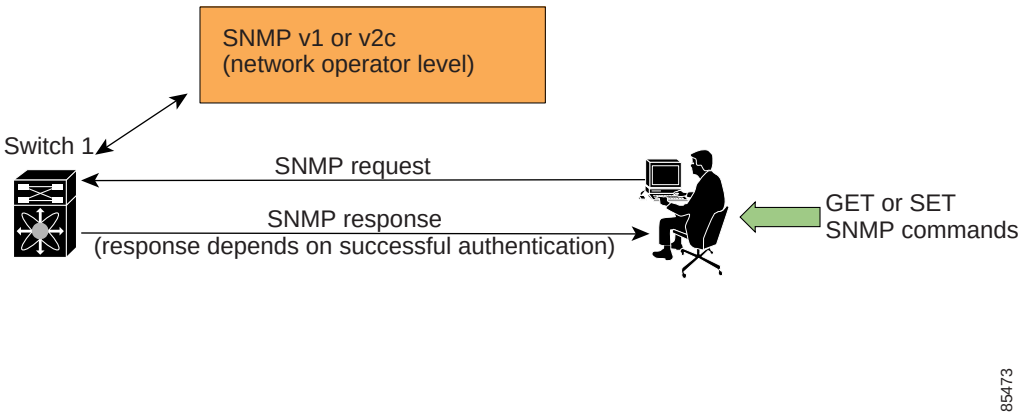
Example 16-18 Displays Host Key Pair Details

```
switch# show ssh key
rsa1 Keys generated:Sun Jan 13 07:16:26 1980
1024 35
fingerprint:
1024 67:76:02:bd:3e:8d:f5:ad:59:5a:1e:c4:5e:44:03:07
could not retrieve rsa key information
dsa Keys generated:Sun Jan 13 07:40:08 1980
ssh-dss AAAAB3NzaC1kc3MAAABBAJTCRQOydNRel2v7uiO6Fix+OTn8eGdnnDVxw5eJs5OcOEXOyjaW
cMMYsEgxc9ada1NElp8Wy7GPMWGOQYj9CU0AAAAVAMCcWhNN18zFNOIPo7cU3t7d0iEbAAAAQBdQ8UAO
i/Cti84qFb3kTqXlS9mEhdQUo0lHcH5bw5PKfj2Y/dLR437zCBKXetPj4p7mhQ6Fq5os8RZtJEyOsNsA
AABAA0oxZbPyWeR5NHATXiyXdPI7j9i8fgyn9FNipMkOF2Mn75Mi/lqQ4NIq0gQNvQOx27uCeQlRts/Q
wI4q68/eaw==
fingerprint:
512 f7:cc:90:3d:f5:8a:a9:ca:48:76:9f:f8:6e:71:d4:ae
```

SNMP Security

SNMP is an application layer protocol that facilitates the exchange of management information between network devices. In all Cisco MDS 9000 Family switches, three SNMP versions are available: SNMPv1, SNMPv2c, and SNMPv3 (see Figure 16-2).

Figure 16-2 SNMP Security



Note Users and roles configured through the CLI are different from users and roles configured through SNMP. These configurations do not directly correspond with each other. However, you can configure both CLI and SNMP identically, if required.

SNMP users are different from CLI users. SNMP users also have role-based authentication for roles and authorization purposes.

SNMP Version 1 and Version 2c

SNMPv1 and SNMPv2c use a community string match for user authentication. Community strings provided a weak form of access control in earlier versions of SNMP. SNMPv3 provides much improved access control using strong authentication and should be preferred over SNMPv1 and SNMPv2c wherever it is supported.

SNMP Version 3

SNMPv3 is an interoperable standards-based protocol for network management. SNMPv3 provides secure access to devices by a combination of authenticating and encrypting frames over the network. The security features provided in SNMPv3 are:

- Message integrity—Ensures that a packet has not been tampered with in-transit.
- Authentication—Determines the message is from a valid source.
- Encryption—Scrambles the packet contents to prevent it from being seen by unauthorized sources.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy that is set up for a user and the role in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

Restricting Switch Access

You can restrict access to a Cisco MDS 9000 Family switch using IP Access Control Lists (IP-ACLs). See the "IP Access Control Lists" section on page 20-5.

Group-Based SNMP Access

Note Because *group* is a standard SNMP term used industry-wide, we refer to role(s) as group(s) in this SNMP section.

SNMP access rights are organized by groups. Each group in SNMP is similar to a role through the CLI. Each group is defined with three accesses: read access, write access, and notification access. Each access can be enabled or disabled within each group.

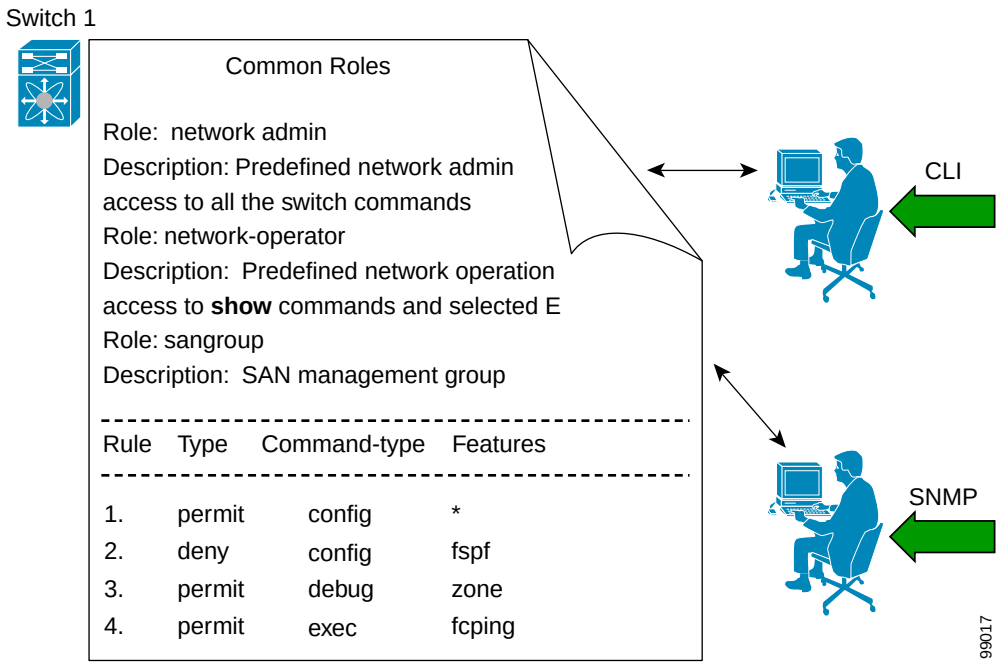
You can begin communicating with the agent once your user name is created, your roles are set up by your administrator, and you are added to the roles.

Note Users configured through the CLI are different from users configured through SNMP. These configurations do not directly correspond with each other. However, you can configure both CLI and SNMP identically, if required.

Configuring Common Roles

From Cisco MDS SAN-OS Release 1.2, CLI and SNMP in all switches in the Cisco MDS 9000 Family use common roles. You can use SNMP to modify a role that was created using CLI and vice versa (see Figure 16-3).

Figure 16-3 Common Roles



Each role in SNMP is the same as a role created or modified through the CLI (see the "Role-Based CLI Authorization" section on page 16-18).

Each role can be restricted to one or more VSAN as required.

You can create new roles or modify existing roles using SNMP or the CLI.

- SNMP—Use the CISCO-COMMON-ROLES-MIB to configure or modify roles. Refer to the *Cisco MDS 9000 Family MIB Quick Reference*.
- CLI—Use the **role name** command.

Creating and Modifying Users

You can create users or modify existing users using SNMP or the CLI.

- SNMP—Create a user as a clone of an existing user in the vsmUserTable on the switch. Once you have created the user, change the cloned secret key before activating the user. Refer to RFC 2574.

Note You must explicitly configure password(s) for SNMP users. The SNMP user passwords are not generated as the part of the configuration file as they are not portable across devices. The password is limited to a minimum of 8 characters and a maximum of 64 characters.

Tip An SNMP user must be created on each switch to which the user requires access. If the user is managing 10 switches, each of the 10 switches must have the SNMP user defined.

- CLI—Create a user or modify an existing user using the **snmp-server user** command.

Caution Cisco MDS SAN-OS does not support all numeric usernames, whether created with TACACS+ or RADIUS, or created locally. Local users with all numeric names cannot be created. If an all numeric username exists on an AAA server and is entered during login, the user is not logged in.

By default only two roles are available in a Cisco MDS 9000 Family switch—network-operator and network-admin. You can also use any role that is configured in the Common Roles database (see the "Configuring Common Roles" section on page 16-32).

Configuring SNMP Users from the CLI

To create or modify SNMP users from the CLI, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# snmp-server user joe network-admin auth sha abcd1234`	Creates or modifies the settings for a user (joe) in the network-admin role using the HMAC-SHA-96 authentication password (abcd1234). **Note** User login IDs must contain non-numeric characters.
	`switch(config)# snmp-server user sam network-admin auth md5 abcdefgh` `switch(config)#`	Creates or modifies the settings for a user (sam) in the network-admin role using the HMAC-MD5-96 authentication password (abcdefgh). **Note** User login IDs must contain non-numeric characters.
	`switch(config)# snmp-server user Bill network-admin auth sha abcd1234 priv abcdefgh`	Creates or modifies the settings for a user (network-admin) in the network-admin role using the HMAC-SHA-96 authentication level and privacy encryption parameters. **Note** User login IDs must contain non-numeric characters.
	`switch(config)# no snmp-server user usernameA`	Deletes the user (usernameA) and all associated parameters.
	`switch(config)# snmp-server user user1 network-admin auth md5 0xab0211gh priv 0x45abf342 localizedkey`	Specifies the password to be in localized key format (see RFC 2574). The localized key is provided in the hex format (for example, 0xacbdef).

Note Avoid using the **localizedkey** option when configuring an SNMP user from the CLI. The localized keys are not portable across devices as they contain device engine ID information. If a configuration file is copied to the device, the passwords may not be set correctly if the configuration file was generated at a different device. Explicitly configure the desired passwords after copying the configuration into the device.

Forcing Identical SNMP and CLI Passwords

You can force the SNMPv3 password and the CLI password to be the same. You must know the SNMPv3 password to change the password using the CLI. Use the CLI password to synchronize the SNMP password The password is limited to a minimum of 8 characters and a maximum of 64 characters.

Caution To change the SNMP password, a clear text CLI password is required.

To modify the secret key for an SNMPv3 user, refer to RFC 2574.

To update the SNMPv3 password from the CLI, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# username joe password wxyz6789 update-snmpv3 abcd1234`	Updates the SNMPv3 password for the specified user (joe). The local CLI password and the SNMP password are updated. If user Joe does not exist, the command fails.

Assigning Users to Roles

Once the user and the role are created, the administrator should configure an entry in the vacmSecurityToGroupTable to add the configured user to a configured role.

To assign users to roles through SNMP, refer to RFC 2575.

To assign users to roles through the CLI, refer to the procedure specified in the "Creating and Modifying Users" section on page 16-32.

Adding or Deleting Communities

You can configure read-only or read-write access for SNMP users by using the **snmp-server community** CLI command. Use the **no** form of the command to delete the configured community. Refer to RFC 2576.

To create an SNMPv1 or SNMPv2c community, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.

	Command	Purpose
Step 2	`switch(config)# snmp-server community snmp_Community ro`	Adds read-only access for the specified SNMP community.
	`switch(config)# snmp-server community snmp_Community rw`	Adds read-write access for the specified SNMP community.
	`switch(config)# no snmp-server community snmp_Community`	Deletes access for the specified SNMP community (default).

Assigning SNMP Switch Contact Information

Use the **snmp-server** command to set the contact information and the switch location. They are each limited to 32 characters (without spaces). Use the **no** form of the command to remove the system contact information.

To configure contact information, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# snmp-server contact NewUser`	Assigns the contact name for the switch.
	`switch(config)# no snmp-server contact NewUser`	Deletes the contact name for the switch.
Step 3	`switch(config)# snmp-server location SanJose`	Assigns the switch location.
	`switch(config)# no snmp-server location SanJose`	Deletes the switch location.

Configuring SNMP Traps

You can configure the Cisco MDS switch using the CLI to send notifications to SNMP managers when particular events occur. You can send these notifications as traps or inform requests. Traps are unreliable because the receiver does not send any acknowledgment when it receives a trap. The sender cannot determine if the trap was received. However, an SNMP manager that receives an inform request acknowledges the message with an SNMP response PDU. If the sender never receives a response, the inform request can be sent again. Thus, informs are more likely to reach their intended destination.

Note Use the SNMP-TARGET-MIB to obtain more information on trap destinations and inform requests. Refer to the *Cisco MDS 9000 Family MIB Quick Reference* for more information.

Tip The SNMP version **1** option is not available with the **snmp-server host ip-address informs** command.

To configure SNMP traps, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.

	Command	Purpose
Step 2	`switch(config)# snmp-server host 171.71.187.101 traps version 2c private udp-port 1163`	Configures the specified host to receive SNMP version 2c trap notifications on a private port number 1163.
	`switch(config)# no snmp-server host 172.18.2.247 informs version 3 public udp-port 2162`	Prevents the specified host to receive SNMP version 3 inform notifications on a private port number 2162.
	`switch(config)# snmp-server host 10.1.1.1 fsdf`	Configures the specified host to receive SNMP inform notifications with the default **noauth** option on the default UDP port (162).

Displaying SNMP Security Information

Use the **show snmp** commands to display configured SNMP information (see Example 16-19 and 16-21).

Example 16-19 Displays SNMP User Details

```
switch# show snmp user
User                          Group                         Auth  Priv
____                          _____                         ____  ____
steve                         network-admin                 md5   des
sadmin                        network-admin                 md5   des
stever                        network-operator              md5   des
```

Example 16-20 Displays SNMP Community Information

```
switch# show snmp community
Community                      Access
---------                      ------
private                         rw
public                          ro
v93RACqPNH                    ro
```

Example 16-21 Displays SNMP Host Information

```
switch# show snmp host
Host                           Port Version  Level  Type   SecName
____                           ____ _______  ______ ____   _______
171.16.126.34                  2162 v2c      noauth trap   public
171.16.75.106                  2162 v2c      noauth trap   public
...
171.31.58.97                   2162 v2c      auth   trap   public
...
```

Displaying SNMP Counter Information

The **show snmp** command displays counter information for SNMP contact, location, and packet settings. This command provides information that is used entirely by the Cisco MDS 9000 Family Fabric Manager (refer to the *Cisco MDS 9000 Family Fabric Manager Configuration Guide*). See Example 16-22.

Example 16-22 Displays SNMP

```
switch# show snmp
switch# show snmp
sys contact:
sys location:
1631 SNMP packets input
        0 Bad SNMP versions
        0 Unknown community name
        0 Illegal operation for community name supplied
        0 Encoding errors
        64294 Number of requested variables
        1 Number of altered variables
        1628 Get-request PDUs
        0 Get-next PDUs
        1 Set-request PDUs
152725 SNMP packets output
```

```
        0 Too big errors
        1 No such name errors
        0 Bad values errors
        0 General errors
Community                      Access
---------                      ------
public                          rw
User                          Group                          Auth  Priv

____                          _____                          ____  ____
admin                         network-admin                  md5   no
```

Default Settings

Table 16-1 lists the default settings for all security features in any switch.

Table 16-1 Default Security Settings

Parameters	Default
Roles in each switch (for CLI and SNMP users)	Two default roles—network-operator and network-admin.
AAA authentication login	Local authentication is enabled. If the Telnet or SSH options are not specified, the command applies to both.
Telnet server	Enabled.
Accounting log file size on local disk	15,000 bytes.
User's account expiration	Does not expire unless you explicitly configure it to expire.
User name	Admin.
User password	Admin.
Configured RADIUS sever	Allows access to all RADIUS severs.
RADIUS server timeout interval	The default timeout is one (1) second.
RADIUS preshared key	No key is configured.
RADIUS key encryption	Clear text (0)—not encrypted.
RADIUS server connection attempts	A switch tries to connect to a RADIUS server once (1).
RADIUS Authentication port	UDP port 1812.
RADIUS Accounting port	UDP port 1813.
Server key encryption	Clear text (0)—not encrypted.
TACACS+	Disabled.
Configured TACACS+ sever	Allows access to all TACACS+ severs.
TACACS+ server timeout interval	The default timeout is one (5) seconds.
TACACS+ preshared key	No key is configured.
TACACS+ key encryption	Clear text (0)—not encrypted.
TACACS+ server connection attempts	A switch tries to connect to a TACACS+ server once (1).
TACACS+ Authentication port	UDP port 49.
VSAN policy	Permit.