CHAPTER 2

Infrastructure Device Access

Securing the network infrastructure itself is critical to overall network security, be they routers, switches, servers, or other infrastructure devices. One key element of this is the security of management access to these infrastructure devices. If infrastructure device access is compromised, the security and management of the entire network can be compromised. Consequently, it is critical to establish the appropriate controls in order to prevent unauthorized access to infrastructure devices.

Network infrastructure devices often provide a range of different access mechanisms, including console and asynchronous connections, as well as remote access based on protocols such as Telnet, rlogin, HTTP, and SSH. Some mechanisms are typically enabled by default with minimal security associated with them; for example, Cisco IOS software-based platforms are shipped with console and modem access enabled by default. For this reason, each infrastructure device should be carefully reviewed and configured to ensure only supported access mechanisms are enabled and that they are properly secured.

The key steps to securing both interactive and management access to an infrastructure device are:

- **Restrict Device Accessibility**

  Limit the accessible ports, restrict the permitted communicators and restrict the permitted methods of access.

- **Present Legal Notification**

  Display legal notice, developed in conjunction with company legal counsel, for interactive sessions.

- **Authenticate Access**

  Ensure access is only granted to authenticated users, groups, and services.

- **Authorize Actions**

  Restrict the actions and views permitted by any particular user, group, or service.

- **Ensure the Confidentiality of Data**

  Protect locally stored sensitive data from viewing and copying. Consider the vulnerability of data in transit over a communication channel to sniffing, session hijacking and man-in-the-middle (MITM) attacks.

- **Log and Account for all Access**

  Record who accessed the device, what occurred, and when for auditing purposes.

Note It is critical to regularly review logs in order to audit access and identify any anomalous access attempts or actions.

CSF Methodology Assessment

The results of applying CSF to baseline infrastructure device access security are presented in Table 2-1 and Table 2-2. The tables highlight the technologies and features identified for baseline secure device management access and that are integrated in Network Security Baseline.

Total Visibility

Table 2-1 CSF Methodology Assessment—Total Visibility

Identify	Monitor	Correlate
• AAA Enforcement – Centralized AAA and local fallback – Administrator access – Privileged level access • SNMP Accounts – Community strings or auth/privacy policy • AAA server definitions • Device Management Best Common Practices – Strong password policy – Per-user accounts – Remove default accounts and passwords	• Logging – Syslog – SNMP – AAA Server Based Accounting – Configuration change notification and logging	

Complete Control

Table 2-2 CSF Methodology Assessment—Total Control

Harden	Isolate	Enforce
• SNMP • SSH/Telnet • HTTP/HTTPS • Restrict Device Accessibility – Transport types – VTY ACLs – SNMP ACLs – IOS login enhancements	• Console • Dedicated management interface • Management Network • SSH/Telnet • HTTP/HTTPS • ACLs • Out-of-band (OOB) Management	• Banners – MOTD – EULA • Local Password Protection – Password encryption – Secrets • File Transfer & Verification – FTP – TFTP – SCP – IOS image verification • Session Management • Device Management Best Common Practices – Minimum access privileges

Restrict Infrastructure Device Management Accessibility

The first step in securing management access to infrastructure devices is to restrict device accessibility. The key elements include:

- Restrict access to authorized terminal and management ports only
- Restrict access to authorized services and protocols only
- Restrict access attempts to authorized services by authorized originators only
- Only grant access to authenticated and authorized users
- Grant minimum privilege levels to authorized users
- Enforce session management
- Restrict vulnerability to dictionary and DoS attacks

The general approach to achieving each of these objectives is listed in Table 2-3. The general philosophy is that device management access should be implicitly denied and only permitted for those users and services that are explicitly required.

Table 2-3 Approaches for Infrastructure Device Management Accessibility Restriction

Infrastructure Device Management Accessibility Restriction	General Approach
Restrict access to authorized terminal and management ports only	• Disable all device terminal and management ports that are not explicitly required or actively being used for device management access
Restrict access to authorized services and protocols only	• Permit device management access only through required and supported services and protocols • Deny all other device management access services and protocols • Deny outgoing access unless explicitly required
Restrict access attempts to authorized services by authorized originators only	• Only permit access attempts to authorized services from authorized originators
Only grant access to authenticated and authorized users	• Use AAA to authenticate and authorize device management access on all supported ports and services
Enforce session management	• Enforce idle timeouts and keepalives to detect and close inactive or hung sessions. • Enforce an active session timeout to restrict the maximum duration of a session prior to re-authentication
Restrict vulnerability to dictionary and DoS attacks	• Limit the rate of login attempts • Enforce a lockout period upon multiple authentication failure attempts • Reserve one management port for access only by one particular NoC host

Note Most infrastructure devices can be accessed through a variety of terminal and management ports, services and protocols, some of which may be enabled by default. All possible management access mechanisms should be reviewed and secured.

Cisco IOS Device Interactive Terminal and Management Access Lines

Cisco IOS software-based platforms typically offer interactive device management access through the following ports and lines:

- TTY lines

  Asynchronous ports, including:

  - AUX
  - console

- VTY lines

Virtual TTY lines used for remote access such as:

- Telnet
- SSH
- rlogin

Note that web-based GUI (HHTP/HTTPS) and SNMP access are covered in subsequent sections.

AUX Port

Interactive access via an AUX port is typically used to provide either dial-in or dial-out management access to a platform. If this is not required, the line should be disabled to reduce the risk of unauthorized access.

Console Port

Interactive access via a console port is directly accessible by a local user or remotely accessible through the use of a terminal or console server. If console port access is required, the line should be properly secured to prevent unauthorized access.

Note If a terminal server is employed, it is vital to ensure that this device is properly secured by enforcing the security guidelines presented in this paper.

VTY Line

Interactive access via a VTY line is the most commonly used method to remotely manage a device. If VTY access is required, the lines should be properly secured to prevent unauthorized access.

Sample TTY and VTY configurations are provided in Appendix A, "Sample Configurations."

Note A router typically has 5 VTY lines (VTY 0 4) but more may be supported. It is critical to ensure that security guidelines are applied to all available VTY lines.

Disable Unnecessary Device Terminal and Management Access Ports

Some network infrastructure devices have terminal and management ports and interfaces enabled by default. This can present a security risk. It is recommended to disable all terminal and management ports and interfaces which are not required or are not used.

On a Cisco IOS device, terminal and management ports typically include TTY and VTY lines. These ports can be disabled using the **no exec** command as shown in the following configuration:

```
!
! Disable access to VTY
line vty 1
 login
 no exec
!
! Disable access to Console
line con 0
  no exec
```

!

Restrict Device Access to Authorized Services and Protocols Only

Some network infrastructure devices have device management access services and protocols enabled by default. This can present a security risk. It is recommended to disable all device management access services and protocols that are not required or are not used.

On a Cisco IOS device, device management access services and protocols typically include:

- Interactive access via Telnet, SSH, etc.
- HTTP, HTTPS
- SNMP

Interactive access, through the TTY and VTY lines of a Cisco IOS device, should be restricted to only those authorized access services and protocols required and permitted, per corporate security policy. Restrictions should be enforced on both incoming and outgoing connections.

This is enforced on TTY and VTY lines using the **transport** command. Some examples are provided in Table 2-4.

Table 2-4 Examples of Restricting Incoming and Outgoing Connections

Cisco IOS TTY and VTY Line Device Management Access Protocol Restriction	IOS Configuration on TTY or VTY
No incoming connections	`transport input none`
No outgoing connections	`transport output none`
Only SSH permitted for incoming connections	`transport input ssh`
Only telnet permitted for incoming connections	`transport input telnet`
SSH or telnet permitted for incoming connections	`transport input telnet ssh`
Only SSH permitted for outgoing connections	`transport output ssh`
Transport protocol must be specified in access request	`transport preferred none`

Note The best practice is to prefer encrypted access protocols, such as SSH, over clear text protocols like Telnet.

SSH is covered in more detail in Secure Shell (SSH), page 2-18.

Security guidelines for HTTP, HTTPS, and SNMP are described at following:

- HTTP, page 2-19
- HTTPS, page 2-20
- SNMP Access, page 2-21

For more information on the **transport** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/termserv/command/reference/ter_t1g.html#wp1083564

Restrict Device Access Attempts To Authorized Services By Authorized Originators Only

Only authorized originators should be permitted to even attempt device management access, and only to the services they are authorized to use. This ensures that the processing of access requests is restricted to only authorized services by an authorized source IP address.This reduces the risk of unauthorized access and the exposure to other attacks, such as brute force, dictionary, or DoS attacks.

On a Cisco IOS device, standard ACLs can be used to restrict device management access attempts to authorized originators only. Extended ACLs can be used to restrict device management access attempts to authorized services by authorized originators only.

It should be noted that the more restrictive an ACL, the more limited the exposure to unauthorized access attempts. However, a more restrictive ACL, such as one restricting access to NoC hosts only, can create a management overhead, can impact accessibility if network connectivity is lost for a specified location and can limit the ability of authorized administrators to perform troubleshooting, such as those at a remote location investigating local anomalies. Consequently, there is a balance to be considered. One compromise is to restrict access to internal corporate IP addresses only.

Note Each customer must evaluate the implementation of ACLs in relation to their own security policy, risks, exposure, and acceptance thereof.

The use of ACLs to restrict traffic directed to an infrastructure device itself is covered in more detail in the Appendix 4, “Device Resiliency and Survivability.”

Standard ACLs

Standard ACLs allow restrictions to be enforced on the originator source IP address or IP address range.

```
!
access-list 10 permit <NOCsubnet> <inverse-mask>
access-list 10 deny any any
!
```

Extended ACLs

Extended ACLs allow restrictions to be enforced on the originator source IP address or IP address range, and the access protocol.

```
!
access-list <xACL#> permit tcp <NOCsubnet1> <inverse-mask> any eq <TCP port>
access-list <xACL#> permit tcp <NOCsubnet2> <inverse-mask> any eq <TCP port>
access-list <xACL#> deny ip any any log-input
!
```

Note Access-class ACLs only support the any clause as destination.

The ACL must subsequently be configured on the appropriate lines, services, and interfaces in order for it to be enforced. An example of how to enforce an ACL on VTY lines is shown below:

```
!
line vty 0 4
 access-class <ACL#> in
!
```

Note A highly restrictive ACL can also be applied to one VTY in order to try to preserve interactive access during a DoS attack on VTY lines. For more information, refer to Restrict Login Vulnerability to Dictionary and DoS Attacks, page 2-10.

ACL enforcement for SNMP and HTTP access is addressed in their related sections (see SNMP Access, page 2-21 and HTTP, page 2-19).

Enforce Device Login Authentication Using AAA

Access to all infrastructure device management ports should be authenticated to restrict access to authorized users only. It is recommended that a centralized AAA server be deployed to enforce per-user, AAA-based login authentication on all infrastructure device terminal and management ports.

In Cisco IOS, administrative access to a network infrastructure device is referred to as an EXEC session and is performed over a TTY or VTY line. AAA-based authentication of EXEC user login is enforced by applying a AAA method list to all available TTY and VTY lines.

An example IOS configuration for the enforcement of AAA-based authentication, with local fallback, for EXEC user login on the console and VTY lines is shown below:

```
!
aaa authentication login adminAuthen-list group adminAAAgroup local-case
!
line con 0
 login authentication adminAuthen-list
!
line vty 0 4
 login authentication adminAuthen-list
!
```

For more information on Cisco IOS named method lists, refer to AAA Method Lists, page 2-16.

For more information on the AAA authentication login command, refer to the following URL: http://www.cisco.com/en/US/docs/ios/12_2/security/configuration/guide/scfathen.html#wp1001032

Enforce Device Login Authorization Using AAA

Only minimum access privileges should be granted to authenticated users, according to their specific access requirements. This reduces exposure to both malicious and unintentional security incidents.

In Cisco IOS, the ability to control which users are authorized to open an EXEC session and access the CLI is achieved using the **aaa authorization exec** command. In conjunction with AAA, different user groups can easily be granted different access privileges.

An example of AAA-based EXEC session access authorization on VTY lines is shown below, including fallback to authorization being granted if a user is authenticated, in case a AAA server is not available.

```
!
aaa authorization exec adminAuthor-list group adminAAAgroup if-authenticated
line vty 0 4
 authorization exec adminAuthor-list
!
```

Note The AAA server must set the ‘Service-Type’ attribute to EXEC (login) in order to grant EXEC session access.

For more information on the **aaa authorization exec** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/security/command/reference/sec_a1g.html#wp1071720

Enforce Privileged Level Authentication Using AAA

It is critical to ensure that an administrative user attempting to obtain privileged level access is properly authenticated. Privileged level access typically refers to a level of access which provides the ability to configure a network infrastructure device.

In Cisco IOS, privileged level EXEC access may be obtained either from the CLI using the **enable** command, or automatically as a result of RADIUS or TACACS+ authorization. The later requires exec authorization and the configuration of the privilege level at the user or group profile on the AAA server.

Privilege level access extends the access level of an EXEC session, providing the ability to configure the device. Therefore, in accordance with the enforcement of minimum access privileges, enable access should only be granted to those users requiring this level of access.

Cisco IOS enable access should be authenticated using AAA-based authentication to a centralized AAA server with local fallback to the enable secret. This is achieved by defining a default AAA method list for enable authentication.

```
!
aaa authentication enable default group adminAAAgroup enable
!
```

It is recommended that an **enable secret** be configured instead of an **enable password**, since the enable secret provides Type 5 encryption which is not reversible. See Enable Secret, page 2-24, for more details.

For more information on the **aaa authentication enable default** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/11_3/security/configuration/guide/scauthen.html#wp6292

Enforce Session Management

Device access sessions should be managed to ensure that the following scenarios are addressed:

- Idle sessions
- Hung sessions

Idle Sessions

Idle sessions should not be permitted to consume a terminal or management port indefinitely. This preserves the availability of terminal and management ports, and reduces the exposure to session hijacking.

In Cisco IOS, an idle timeout is configurable on TTY and VTY lines with the command session-timeout. By default, a VTY session has a 10 minute idle timeout.

```
Router(config-line)# session-timeout <minutes>
```

The session-timeout command behaves slightly differently on virtual (VTY) terminals than on physical console, auxiliary (AUX), and terminal (TTY) lines. When a timeout occurs on a VTY, the user session returns to the EXEC prompt. When a timeout occurs on physical lines, the user session is logged out and the line returned to the idle state.

You can use a combination of the exec-timeout and session-timeout line configuration commands, set to approximately the same values, to get the same behavior from virtual lines that the session-timeout command causes on physical lines.

In Cisco IOS, by default a VTY session has a 10 minute exec timeout.

```
Router(config-line)# exec-timeout <minutes> [seconds]
```

For more information on the **session-timeout** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/termserv/command/reference/ter_l1g.html#wp1037637

For more information on the **exec-timeout** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g03.html#wp1029531

Hung Sessions

If a remote system crashes while a management session is in progress, the session may remain open and vulnerable to attack. Consequently, hung sessions should be detected and closed in order to preserve the availability of terminal and management ports and reduce the exposure to session hijacking.

In Cisco IOS, hung sessions on VTY lines can be detected and closed with the **service tcp-keepalives-in** command. This causes TCP keepalives to be sent on incoming connections, enabling a remote system crash to be detected if no response is received.

```
Router(config)# service tcp-keepalives-in
```

For more information on the **service tcp-keepalives-in** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g07.html#wp1029289

Restrict Login Vulnerability to Dictionary and DoS Attacks

The vulnerability of a network infrastructure device management access to dictionary and DoS attacks may be reduced by enforcing the following restrictions:

- Enforce the use of strong passwords
- Restrict the frequency of login attempts
- Restrict the number of login failures permitted within a specified time period
- Reserve a terminal or management port

The features available to enforce these requirements are listed Table 2-5.

Table 2-5 Features for Restricting Login Vulnerability to Dictionary and DoS Attacks

Requirement	Implementation Options
Enforce the use of strong passwords	AAA server feature: Enforce the use of strong passwords on the AAA server, in compliance with the security policy
	Cisco IOS feature to force a minimum password length[1]: `security passwords min-length`
Restrict the frequency of login attempts	Cisco IOS login enhancement feature[2]: `login delay`
Restrict the number of login failures permitted within a specified time period	AAA server feature: Enforce account lockout on the AAA server if a defined number of failed login attempts within a specified time period is exceeded

	Cisco IOS login enhancements [2, 3]
	`login block-for`
	`login quiet-mode access-class`

Reserve one terminal or management port	Cisco IOS feature: Highly restrictive ACL on last VTY line
Log and monitor user login authentication failures	See Infrastructure Device Management Access Logging, page 2-24.

1. The Cisco IOS feature to force a minimum password length was introduced in 12.3(1) and was integrated into Cisco IOS software Release 12.2(18)T.
2. The Cisco IOS login enhancement feature was introduced in 12.3(4)T and 12.2(25)S.
3. The Cisco IOS login enhancement feature to restrict the number of login failures permitted within a specified time period is typically only used when a AAA server is not being employed to enforce authentication.

For more information about Cisco IOS login enhancements, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/configuration/guide/sec_login_enhance_ps6350_TSD_Products_Configuration_Guide_Chapter.html

Enforce The Use of Strong Passwords

In the event of a dictionary attack, the use of strong passwords makes such an attack less likely to succeed, since the passwords will not be simple dictionary words. If a AAA server is employed for login authentication, the AAA server typically offers a feature to enforce the use of strong passwords, according to the security policy. If a AAA server is not available to enforce a strong password policy, local device features to minimize vulnerability to dictionary attacks should be employed, as available.

If features to address password vulnerability to dictionary attacks are not available, a basic feature that may be available is the enforcement of a minimum password length. Whilst this type of feature does not provide direct protection against dictionary attacks, it provides protection against the use of commonly guessed passwords such as **cisco** and **lab**.

Cisco IOS Minimum Password Length Feature

Cisco IOS offers the ability to enforce a minimum password length for user passwords, enable passwords, enable secrets, and line passwords. This feature is enabled with the global configuration command:

```
Router(config)# security passwords min-length length
```

Once this command is enabled, any password that is less than the specified number of characters will fail.

Note This feature does not provide any protection against dictionary attacks.

For more information about the **security passwords min-length** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/security/command/reference/sec_r1g.html#wp1081544

Restrict Frequency of Login Attempts

In the event of a dictionary attack, introducing a delay between login attempts slows down the attack, increasing the time required for the attack to succeed and the timeframe available for the anomaly to be identified and addressed.

In Cisco IOS, the introduction of a delay between successive login attempts can be achieved using the global configuration **login delay** command. The default is a 1 second delay.

```
Router(config)# login delay <seconds>
```

For more information on the **login-delay** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_k1.html#wp1031384

Restrict Number of Login Failures Permitted Within Specified Time Period

In the event of a dictionary attack, restricting the maximum number of failed access attempts within a specified period can slow down the attack, increasing the time required to succeed and the timeframe available for the anomaly to be identified and addressed.

If a AAA server is employed for login authentication, the AAA server typically offers a feature, enforcing account lockout if a defined number of failed login attempts occur within a specified time period. If a AAA server is not employed, then the Cisco IOS feature may be employed.

In Cisco IOS, the definition of the maximum number of failed login attempts permitted within a specified time period, after which the IOS device will not accept any additional connection attempts for a configurable "quiet period", can be achieved using the global configuration **login block-for** command as follows:

```
Router(config)# login block-for seconds attempts tries within seconds
```

For more information on the **login block-for** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_k1.html#wp1031219

The Cisco IOS also offers the ability to define an exception ACL for trusted systems and networks from which legitimate connections are expected. This exception ACL can be defined with the **login quiet-mode access-class** global command:

```
Router (config)# login quiet-mode access-class
```

For more information on the **login quiet-mode access-class** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_k1.html#wp1031806

The following example shows how to configure a router to enter a 100-second quiet period if 15 failed login attempts is exceeded within 100 seconds; all login requests will be denied during the quiet period except from the host defined in ACL 10.

```
Router(config)# access-list 10 permit host 172.26.150.206
Router(config)# login block-for 100 attempts 15 within 100
Router(config)# login quiet-mode access-class 10
```

Reserve a Terminal or Management Port

A DoS attack on an infrastructure device can target the terminal and management ports. This type of attack relies on the fact that there are only a limited number of terminal and management ports available and that, once all ports are in use, even if the connection has not yet been authenticated, no additional connections can be established.

Cisco IOS software devices have only a limited number of VTY lines, typically five. When all VTY lines are in use, no more remote interactive connections can be established. This creates an opportunity for a DoS attack if an attacker can open remote sessions to all VTYs available on a system, preventing an authorized administrator from gaining access. The attacker does not need to log in to achieve this type of DoS attack, the remote sessions can simply be left at the login prompt. The use of AAA does not mitigate this type of attack as the attacker does not need to attempt a login, it is only necessary to maintain a connection to the port, thus rendering it unavailable to other users.

One way to address this type of attack is to enforce highly restrictive access controls on one terminal or management port to preserve availability during this type of DoS attack. For example, this port can only be accessed by one particular NoC host.

In Cisco IOS, this may be achieved by configuring a highly restrictive ACL on the last VTY. The last VTY, usually VTY 4, can be restricted to accept connections only from a single, specific administrative station, such as a highly secured NoC host, whereas the other VTYs accept connections from any address in a wider address range, such as the NoC.

```
!
access-list 10 permit <NOCsubnet> <inverse-mask>
access-list 20 permit host <NOC-Host>
line vty 0 3
     access-class 10 in
line vty 4
     access-class 20 in
!
```

Legal Notification Banners

It is recommended that a legal notification banner is presented on all interactive sessions to ensure that users are notified of the security policy being enforced and to which they are subject.

In some jurisdictions, civil and/or criminal prosecution of an attacker who breaks into a system is easier, or even required, if a legal notification banner is presented, informing unauthorized users that their use is in fact unauthorized. In some jurisdictions, it may also be forbidden to monitor the activity of an unauthorized user unless they have been notified of the intent to do so.

Legal notification requirements are complex and vary in each jurisdiction and situation. Even within jurisdictions, legal opinions vary, and this issue should be discussed with your own legal counsel to ensure that it meets company, local and international legal requirements. This is often critical to securing appropriate action in the event of a security breach.

In cooperation with the company legal counsel, statements which may be included in a legal notification banner include:

- Notification that system access and use is permitted only by specifically authorized personnel, and perhaps information about who may authorize use.
- Notification that unauthorized access and use of the system is unlawful, and may be subject to civil and/or criminal penalties.
- Notification that access and use of the system may be logged or monitored without further notice, and the resulting logs may be used as evidence in court.
- Additional specific notices required by specific local laws.

From a security, rather than a legal, point of view, a legal notification banner should not contain any specific information about the device, such as its name, model, software, location, operator or owner because this kind of information may be useful to an attacker.

A sample legal notification banner is provided in Appendix A, "Sample Configurations."

In Cisco IOS, a number of banner options are available, including **banner motd**, **banner login**, **banner incoming**, and **banner exe**c.

When a user connects to an IOS device, a message-of-the-day (MOTD) banner, if configured, will appear first, followed by a login banner (if configured) and a login prompt. After a user successfully logs in to an IOS device, an incoming banner will be displayed for a reverse Telnet login and an EXEC banner will be displayed for all other types of connections.

It is recommended that either a MOTD or a login banner is implemented to ensure that a legal notification banner is presented on all device management access sessions, prior to a login prompt being presented.

For more information about the **banner login** or the **banner motd** commands, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g01.html#wp1029811

AAA Services

AAA Overview

AAA is an architectural framework for configuring a set of three independent security functions in a consistent, modular manner:

- **Authentication**

  Enables a user to be identified and verified prior to them being granted access to the network and network services.

- **Authorization**

  Defines the access privileges and restrictions to be enforced for an authenticated user.

- **Accounting**

  Provides the ability to track user access, including user identities, start and stop times, executed commands (such as command line interface (CLI) commands), number of packets, and number of bytes.

AAA is the primary and recommended method for access control. Cisco IOS software provides additional features for simple access control, such as local username authentication and line password authentication, however, these features do not provide the same degree of access control that is possible with AAA and are not recommended, even for small deployments. Even if a separate AAA server is not being deployed, AAA to the local database should used on the Cisco IOS device. See the section on Centralized AAA for more information on the value of AAA to a local database.

AAA authentication, authorization and accounting are enforced by applying named method lists to access interfaces. Method lists are covered in detail in a subsequent section.

AAA uses protocols such as RADIUS, TACACS+, or Kerberos to administer its security functions.

For more information on AAA services, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/configuration/guide/sec_aaa_overview_ps6350_TSD_Products_Configuration_Guide_Chapter.html

Centralized AAA

The recommended method of administering AAA is on a centralized AAA server with local passwords as a fallback method. Local fallback provides a method of authentication in case communication with the AAA server is not possible. The key benefits of using a centralized AAA server include:

- **Manageability**

  Usernames and passwords are stored in a separate, central location which may be independently managed and leveraged across multiple devices.

- **Scalability**

  The AAA server(s) may be independently scaled according to the size of the user database and the number of transactions per second.

- **Security**

  Company-wide usernames and passwords may be stored off the router in a secure, encrypted file system or database. In contrast, locally stored passwords on Cisco IOS devices, even if encrypted, are still reversible.

- **Accountability**

  Access attempts and authorized sessions may be independently logged on the AAA server

If a centralized AAA server is not currently required or deployed, it is still recommended to implement authentication using a AAA configuration, even though a local user password database will be used. This enables the implementation of per-user local passwords, rather than all users using the same login secret or password. This approach offers greater security, visibility and control, along with easier migration to a possible future deployment leveraging a centralized AAA server.

AAA Server Groups

In Cisco IOS, a AAA server group is a list of AAA server hosts of a particular type, e.g. RADIUS or TACACS+, which are used to perform AAA. The particular AAA server group to be used for each particular AAA service is defined by the AAA method list, as discussed below.

The use of the AAA server-group feature provides greater flexibility and control over which AAA servers are used for which purposes, as well as offering redundancy across the defined servers.

For example, different AAA servers may be used for different AAA services to enable the separation and prioritization of device access management from end-user access management through the use of two independently maintained and scaled data stores. For example, infrastructure device access management may be authenticated using a set of TACACS+ servers, whereas end-user network access may be authenticated using a set of RADIUS servers.

```
!
aaa group server tacacs+ adminAAAgroup
 server TAC+server1
 server TAC+server2
!
aaa group server radius enduserAAAgroup
 server RADserver1
 server RADserver2
!
```

For more information on the **aaa group server** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_a1.html#wp1045019

AAA Method Lists

In Cisco IOS, AAA is enforced through the definition of named method lists which are applied to access interfaces. A method list is a sequential list that defines the authentication or authorization methods to be enforced and the sequence in which they will be attempted. Method lists enable one or more security protocols to be used for authentication or authorization, ensuring availability of a backup system in case an initial method is not available.

Cisco IOS software will first attempt the first method listed; if that method does not respond, the next method in the method list will be attempted. This process continues until there is successful communication with a listed method or the method list is exhausted, in which case authentication or authorization fails.

A sample named authentication method list with the name **admin-list** , whose first method is to attempt authentication to the TACACAS+ servers in the server group **adminAAAgroup**, falling back to local authentication if those servers are not available is shown below:

```
aaa authentication login adminAuthen-list group adminAAAgroup local-case
aaa group server tacacs+ adminAAAgroup
 server TAC+server1
 server TAC+server2
```

Note Cisco IOS software attempts authentication or authorization with the next listed method only when there is no response from the previous method. If authentication or authorization fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the process stops and no other methods are attempted.

A AAA method list must be applied to an access line before it will be enforced. The only exception to this is a default AAA method list (which is named **default**). A named method list is automatically applied to all access lines if no other method list is applied. A defined method list overrides the default method list.

```
aaa authentication login default group enduserAAAgroup local-case
aaa group server radius enduserAAAgroup
 server RADserver1
 server RADserver2
```

For more information on AAA method lists and groups, refer to the following URLs:

http://www.cisco.com/en/US/docs/ios/security/configuration/guide/sec_aaa_overview_ps6350_TSD_Products_Configuration_Guide_Chapter.html#wp1000933

http://www.cisco.com/en/US/docs/ios/security/configuration/guide/sec_cfg_radius_ps6350_TSD_Products_Configuration_Guide_Chapter.html#wp1001168

http://www.cisco.com/en/US/docs/ios/security/configuration/guide/sec_cfg_tacacs+_ps6350_TSD_Products_Configuration_Guide_Chapter.html#wp1001011

AAA Server Communication Security

Communication between an authenticator (also referred to as a NAS, Network Access Server) and a AAA server is commonly performed using RADIUS or TACACS+. The security of this communication can be summarized as follows:

- Both RADIUS and TACACS+ transactions are authenticated using a shared, static secret (or key) associated with the device name or IP address but this secret is never sent over the network.
- RADIUS, per the standard, only encrypts the user password field. All other packet data is passed in clear text and is thus vulnerable to sniffing.
- TACACS+ encrypts the full payload of the packet, thereby providing some confidentiality of data, though the encryption algorithm is not very strong.

The general guidelines for securing AAA server communication are:

- Employ strong secrets for authentication of the AAA server and NAS.
- Regularly change the secrets used to authenticate the AAA server and NAS.
- Restrict AAA communication to the limited set of authorized AAA servers, and over the configured AAA communication ports, using extended ACLs.

  The use of ACLs to restrict traffic directed to an infrastructure device itself is covered in more detail in Appendix 4, “Device Resiliency and Survivability.”
- Since RADIUS and TACACS+ do not support strong authentication and encryption, it is recommended that an out-of-band (OOB) or IPSec management network be considered as a means of protecting AAA server communication transactions from attack.

AAA Server Based Accounting Services

It is critical to ensure that device management access is logged. This is covered in more detail in General Device Access and Configuration Change Logging Best Common Practices, page 2-29, but one method of logging device management access is using AAA server based accounting.

AAA server-based accounting enables the ability to track the services users are accessing, as well as the amount of network resources they are consuming. When AAA server-based accounting is enabled, the network infrastructure device reports user activity to the AAA server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the AAA server. This data can then be analyzed for network management, client billing, and/or auditing.

Cisco IOS software supports five different kinds of accounting:

- Network Accounting

  Network accounting provides information for all PPP, SLIP, or ARAP sessions, including packet and byte counts.

- **Connection Accounting**

  Connection accounting provides information about all outbound connections made from the network access server, such as Telnet, local-area transport (LAT), TN3270, packet assembly-disassembly (PAD), and rlogin.

- **EXEC Accounting**

  EXEC accounting provides information about user EXEC terminal sessions (user shells) on the network access server, including username, date, start and stop times, the access server IP address, and (for dial-in users) the telephone number the call originated from.

- **Command Accounting**

  Command accounting provides information about the EXEC shell commands for a specified privilege level that are being executed on a network access server. Each command accounting record includes a list of the commands executed for that privilege level, as well as the date and time each command was executed, and the user who executed it.

- **System Accounting**

  System accounting provides information about all system-level events (for example, when the system reboots or when accounting is turned on or off).

The Network Security Baseline is focused on securing the network infrastructure and critical network services. Consequently, AAA-based accounting in Network Security Baseline includes:

- EXEC accounting
- Command accounting
- System accounting

These elements are covered in detail in Infrastructure Device Management Access Logging, page 2-24.

Secure Shell (SSH)

SSH is a protocol that provides secure remote access, remote command execution, and file transfer. SSH implements strong authentication and encryption, making it a better option over insecure protocols such as rlogin and Telnet.

There are two versions of SSH: v1 and v2. SSHv2 addresses a series of security issues found in the v1. For these reasons, v2 should be used whenever it is supported. Cisco IOS supports both versions of SSH.

SSH authentication supports a variety of protocols including TACACS+, RADIUS, and RSA authentication. SSH also provides support for a wide range of encryption ciphers such as DES, 3DES, IDEA, RC4-128, and others. In addition, SSH can tunnel TCP connections, which allows not only securing login sessions, but also email and file transfers with secure copy (SCP) and secure FTP (SFTP).

The following steps are required to enable SSH support on an IOS device:

Step 1 Configure a hostname and DNS domain for the router.

Step 2 Generate an RSA key pair.

Step 3 Optionally, configure time-out and number of authentication retries. By default, the authentication timeout is set to 120 seconds and authentication retries to three attempts.

Step 4 Limit VTYs to SSH only (Highly recommended).

Step 5 Restrict SSH access to trusted hosts or subnets.

Note On some Cisco IOS platforms, SSH requires an IPSec (DES or 3DES) encryption IOS software image.

The following example shows how SSH can be configured on a Cisco IOS device:

```
!--- Step 1: Configure a hostname and domain name
Router(config)# hostname router
Router (config)# ip domain-name nyc.cisco.com

!--- Step 2: Generate an RSA key pair, automatically enabling SSH.
Router (config)# cry key generate rsa

!--- Step 3: Configure time-out and number of authentication retries.
Router (config)# ip ssh time-out 60
Router (config)# ip ssh authentication-retries 2

!--- Step 4: Configure VTYs to only accept SSH.
Router (config)# line vty 0 4
Router (config-line)# transport input ssh

!--- Step 5: Allow SSH connections only originated from the management network.
Router (config)# access-list 111 remark ACL for SSH
Router (config)# access-list 111 permit tcp 172.26.0.0 0.0.255.255 any eq 22
Router (config)# access-list 111 deny ip any any log-input
Router (config)# line vty 0 4
Router (config-line)# access-class 111 in
```

For more information on restricting which protocols are authorized on device terminal and management ports, see Restrict Device Access to Authorized Services and Protocols Only, page 2-6.

For more information about SSH configuration on IOS routers, refer to the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios124/124cg/hsec_c/part25/ch10/index.htm

Web-based GUI Access

Today almost every networking product can be configured and monitored with a Web-based user interface or GUI (graphical user interface). This type of user interface is popular because it provides a convenient way to access network equipment remotely with the use of a simple web browser.

However, some Web-based user interfaces rely on insecure protocols such as HTTP. HTTP transmits all usernames, passwords and session data in clear text. Consequently, HTTP access is vulnerable to sniffing, interception and other attacks. It is recommended that HTTP access be disabled and that Secure HTTP (HTTPS) be used as an alternative wherever possible. HTTPS uses Secure Sockets Layer (SSL) and Transport Layer Security (TLS) to provide device authentication and data encryption, delivering an acceptable level of protection.

HTTP

In Cisco IOS, HTTP is disabled by default. HTTP may be disabled using the following command:

```
Router(config)# no ip http server
```

For cases where HTTPS is not available as an alternative and HTTP access is absolutely required, the configuration guidelines outlined in Table 2-6 are recommended.

Table 2-6 HTTP Configuration

HHTP Security Guidelines	Cisco IOS Implementation
Authenticate users using AAA in conjunction with a strong password policy	`ip http authentication aaa login-authentication <aaa-listname>`
Authorize HTTP exec commands using AAA	`ip http authentication aaa exec-authorization <aaa-listname>`
Restrict incoming HTTP access attempts from a limited set of authorized HTTP management stations	`ip http access-class <ACL#>` `access-list <ACL#> permit host 10.0.0.1`
Limit the maximum number of concurrent HTTP connections to the expected operational number	`ip http max-connections 3`

In Cisco IOS, HTTP authentication can be enabled with the **ip http authentication** global command. The following example shows a configuration listing for HTTP authentication using TACACS+.

```
!
username adminuser privilege 15 password <mypassword>
!
aaa new-model
aaa authentication login default group adminAAAgroup local-case
aaa authorization exec default group adminAAAgroup local
aaa accounting exec default start-stop group adminAAAgroup
!
ip http server
ip http authentication aaa
!
! HTTP access requires telnet service being accepted at the VTY
line vty 0 4
 transport input telnet
```

For more information about HTTP authentication, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_2/configfun/configuration/guide/fcf005.html#wp1000960

For more details about the **http access-class** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g04.html#wp1028455

For more information about the **ip http max-connections** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g04.html#wp1028838

A sample HTTP configuration is provided in Appendix A, “Sample Configurations.”

HTTPS

In Cisco IOS, HTTPS services can be enabled with the following global configuration command:

```
Router(config)# ip http secure-server
```

For more information on the **ip http secure-server** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g04.html#wp1029339

A sample HTTPS configuration is provided in Appendix A, "Sample Configurations."

SNMP Access

Simple Network Management Protocol (SNMP) is the most popular network management protocol and, as such, is widely supported in the networking industry. SNMP features support for:

- **SNMP read**

  An SNMP manager request for information.

- **SNMP write**

  An SNMP manager request to configure a device.

- **SNMP trap**

  An unsolicited notification sent from an SNMP agent, for example, an infrastructure device, to an SNMP manager.

- **SNMP inform**

  An unsolicited, acknowledged notification sent from an SNMP agent, for example, an infrastructure device, to an SNMP manager. Whilst SNMP informs are more reliable than SNMP traps, they consume more resources and are thus not typically used.

There are three versions of SNMP:

- Version 1, the oldest but still frequently supported
- Version 2c, the most commonly deployed
- Version 3, an IETF standard that provides enhanced security

SNMP versions 1 and 2c are weak in terms of security as these early versions only authenticate access to MIB objects using a community string. In addition, all communication is sent in clear text and is thus vulnerable to sniffing. Neither version supports encryption. Consequently, unauthorized users are able to execute SNMP transactions, and masquerade as legitimate users simply by sniffing and employing the configured community string. In addition, the lack of encryption facilitates the interception of SNMP messages, potentially leading to the disclosure of sensitive information.

In contrast, SNMP v3 addresses some of the security limitations of SNMP v1 and v2c by incorporating security features such as authentication, message integrity check, access controls and encryption. DES encryption enables all communication to be encrypted in order to provide confidentiality of the data. SNMP v3 also provides authorization and access controls, enabling different users to be granted different views, in accordance with the enforcement of minimum access privileges.

Consequently, it is recommended that SNMP v3, with strong authentication and encryption, should be preferred over SNMPv1 or SNMPv2c wherever it is supported due to its enhanced security features.

A summary of the security features of each version of SNMP is provided in Table 2-7.

Table 2-7 SNMP Security Features

Version	Security Level	IOS Keyword	Authentication	Encryption
SNMP v1	noAuthNoPriv	N/A	Community string	°
SNMP v2c	noAuthNoPriv	N/A	Community string	°

Table 2-7 SNMP Security Features (continued)

SNMP v3	noAuthNoPriv	noauth	Username	°
	authNoPriv	auth	MD5 or SHA	°
	authPriv	priv	MD5 or SHA	DES-56

The general guideline for securing SNMP access is to disable SNMP access if it is not required. If SNMP access is required, it is recommended that the following configuration guidelines be considered:

- Selectively use SNMP for required actions only
- Restrict actions to read-only queries

Note Write access creates significant risk and is not recommended.

- Deny queries that request to download the full IP routing and ARP tables using SNMP views
- Treat community strings like root passwords
- Delete default community strings
- Define strong, non-trivial community strings
- Restrict incoming SNMP access attempts to a limited set of authorized SNMP management stations through the use of extended ACLs using recognized SNMP ports (UDP 161 and 162)
- Restrict the SNMP actions permitted by any particular SNMP management station by using different SNMP community strings with different associated SNMP views and different ACLs, permitting only the minimal required level of information to any particular SNMP management station
- If only SNMP v3 is being used, ensure only SNMP v3 access is enabled and with the highest possible security level supported by the communicators
- Enable only operationally important traps (e.g., BGP state changes)
- Send a trap on community name authentication failures to track failed access attempts
- Send a trap for configuration changes and environmental monitor threshold exceptions
- Ensure SNMP traps are regularly monitored

If SNMP v1 or SNMP v2c must be used, it is recommended that IPSec or an out-of-band (OOB) management network are considered as a means of protecting SNMP v1 and SNMP v2c transactions from attack.

A sample SNMP configuration is available in Appendix A, "Sample Configurations."

For more information about SNMPv3, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_2/configfun/configuration/guide/fcf014.html

Locally Stored Information Protection

Cisco IOS devices store some sensitive information locally, including passwords and secrets. Passwords should generally be maintained and controlled by a centralized AAA server. Centralized AAA servers are preferred as they provide a number of features that facilitate secure password management, including

the ability to enforce strong passwords, force users to periodically change their passwords, lock accounts after a specific number of failed login attempts, and many other useful options. In addition, they enable the passwords to be stored and backed up in a secure manner.

However, even if a centralized AAA server is deployed, some locally stored passwords are required on for certain cases, such as local fallback in the case of AAA servers not being available, special-use usernames, secret keys, and other password information. These local passwords and other sensitive information are stored locally in the configuration file.

Cisco IOS offers the following features to enable locally stored sensitive information to be retained in a secure manner in the configuration file:

- Global password encryption
- Local user password encryption
- Enable secret

It should be noted that the encryption of locally stored passwords is employed to preserve the confidentiality of sensitive information stored in a configuration file, both during local viewing of the file and during its transfer. Local password encryption does not address the vulnerability of passwords and secrets to dictionary attacks. See the section Restrict Login Vulnerability to Dictionary and DoS Attacks for more information.

Global Password Encryption

In a Cisco IOS configuration file or listing, by default, some passwords and secrets are stored and presented in clear text. Local passwords and secrets stored in an IOS configuration file should be encrypted to prevent over-the-shoulder browsing of sensitive information.

Cisco IOS offers the ability to encrypt locally stored passwords, CHAP secrets, and similar data in the configuration file. This is enabled using the following global configuration command:

```
Router(config)# service password-encryption
```

However, the encryption algorithm used by the **service password-encryption** command is a simple Vigenere cipher (Type 7) that can be easily reversed. Consequently, this command is primarily only useful for protection from *shoulder surfing*, where an unauthorized individuals attempts to view passwords in a configuration file simply by looking over the shoulder of an authorized user.

Note that not all sensitive data in a configuration file is encrypted by this command. Consequently, configuration files should always be treated as sensitive documents and properly protected to ensure the confidentiality of the data. Particular concern and necessary steps should be taken to protect them during transfer.

Cisco IOS offers support for a stronger encryption algorithm (Type 5) for some locally stored passwords and this should be leveraged whenever available. For example, define local users using the **secret** keyword instead of the **password** keyword, and use **enable secret** instead of **enable password**.

For more information on the **service password-encryption** command, refer to the following URL :

http://www.cisco.com/en/US/docs/ios/12_3/security/command/reference/sec_r1g.html#wp1070450

Local User Password Encryption

Local user names and passwords should be stored in the most secure manner available on a device. These local accounts should only be used as a local fallback method in case the AAA servers for AAA-based authentication are not available.

In Cisco IOS, the passwords for locally configured usernames can be stored using a strong, MD5 encryption algorithm (Type 5) by using the secret keyword instead of the password keyword, where available. For example:

```
Router(config)# username <name> secret <strongpassword>
```

Note that MD5 encryption is not retrievable and thus cannot be used with protocols that require clear text passwords, such as Challenge Handshake Authentication Protocol (CHAP).

For more information on the **username <*name*> secret** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_t2.html#wp1031804

Enable Secret

Local privileged level administrative passwords should be stored in the most secure manner available on a device. Privileged level access typically refers to a level of access which provides the ability to configure a network infrastructure device. Local privileged level administrative passwords should only be used as a local fallback method in case the AAA servers for AAA-based authentication are not available.

In Cisco IOS, privileged level EXEC sessions are accessed using the **enable** command. It is recommended that an **enable secret** be configured instead of an **enable password**, as the **enable secret** uses a strong, MD5 encryption algorithm (Type 5) which is not reversible.

To set an **enable secret**, use the global configuration command:

```
Router(config)# enable secret <strongpassword>
```

By default, an enable secret password is not configured and as a general best practice, one should always be set. If no enable secret is set and a password is configured for the console TTY line, the console password can be used to obtain privileged level access, even from a remote VTY session. Consequently, it is recommended that an enable secret always be configured on a device.

If an enable password is configured, it should be removed to ensure that enable access can only be obtained using the enable secret. An enable password can be removed with the global configuration command:

```
Router(config)# no enable password
```

It should be noted that privileged level access remains vulnerable to dictionary attacks even if an enable secret is employed. For more information on how to address dictionary attacks, see the section Restrict Login Vulnerability to Dictionary and DoS Attacks.

For more information about the **enable secret** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_e1.html#wp1022924

Infrastructure Device Management Access Logging

It is critical to ensure that infrastructure device access and configuration changes are logged to record the following information:

- Who accessed a device
- When a user logged in
- What a user did

- When a user logged off
- Failed access attempts
- Failed authentication requests
- Failed authorization requests

This information is invaluable for forensic analysis in the case of unauthorized attempts or access, as well as for configuration change issues and to help plan group administration changes. It may also be used in real time to identify anomalous activity which may indicate that an attack is taking place. Automated tools are available to enable this analysis and correlate information from additional sources, such as IDS and firewall logs.

The baseline logging objectives and the Cisco IOS features available to achieve them are outlined in Table 2-8.

Table 2-8 Features for Infrastructure Device Management Access Logging

Baseline Logging Objectives	Available Cisco IOS Features
Log successful device access attempts	AAA EXEC accounting
	Syslog login success notification [1]
Log failed device access attempts	AAA failed authentication accounting
	Syslog login failure notification[2]
Log commands entered in EXEC and privilege modes	AAA command accounting
	Archive configuration change logger
Log system-level events, such as system reboots or accounting on/off	AAA system accounting

1. A syslog notification of login success or failure can be used to complement the AAA EXEC accounting records.

AAA EXEC Accounting

Cisco IOS AAA EXEC accounting provides information about user EXEC sessions on the network infrastructure device, including username, date, start and stop times, the device IP address ,and the user source IP address.

A syslog notification of login success or failure can be used to complement the AAA EXEC accounting records.

The following is an example of TACACS+ EXEC accounting records generated for an administrative session. These accounting records were extracted from the TACACS+ Accounting active log of a Cisco ACS server.

Date	Time	User-Name	Group-Name	Caller-Id	Acct-Flags	elapsed_time	service	task_id	NAS-Portname	NAS-IP-Address
2/25/2008	17:11:20	admin2	Group 1	172.26.158.234	start		shell	20	tty1	172.26.158.235
2/25/2008	17:11:41	admin2	Group 1	172.26.158.234	stop	21	shell	20	tty1	172.26.158.235

AAA EXEC accounting is enabled with the aaa accounting exec command. A sample configuration using a named method list and AAA server group, plus its enforcement on VTY lines is shown below:

```
!
aaa accounting exec account-exec-list start-stop group adminAAAgroup
!
line vty 0 4
 accounting exec account-exec-list
```

```
!
```

For more information on the **aaa accounting** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_a1.html#wp1038916

For more information on AAA, see AAA Services, page 2-14.

AAA Failed Authentication Accounting

When AAA accounting is enabled, Cisco IOS software does not generate accounting records for system users who fail login authentication, or who succeed in login authentication but fail PPP negotiation for some reason.

To specify that accounting stop records be generated for users who fail authentication at login or fail PPP session negotiation, use the following command in global configuration mode:

```
Router (config)# aaa accounting send stop-record authentication failure
```

Note This feature does not currently permit the definition of a named AAA method list or a AAA server group. The first RADIUS server configured on the device using **radius-server host** *name* command is the one to which accounting records will be sent.

For more information on the **aaa accounting send stop-record authentication failure** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_a1.html#wp1040064

For more information on AAA, see the AAA Services, page 2-14.

AAA Command Accounting

Available only with TACACS+, Cisco IOS AAA Command accounting provides information about the EXEC session commands for a specified privilege level that are executed on a network infrastructure device. Each command accounting record includes a list of the commands executed for that privilege level, the date and time each command was executed and the user who executed it.

The following is an example of TACACS+ command accounting records generated for an administrative session. These accounting records were extracted from the TACACS+ Administration active log of a Cisco ACS server.

Date	Time	User-Name	Group-Name	cmd	priv-lvl	service	NAS-Portn	task_id	NAS-IP-Address
2/25/2008	17:40:39	admin2	Group 1	configure terminal <cr>	15	shell	tty1	33	172.26.158.235
2/25/2008	17:40:55	admin2	Group 1	hostname c19-6500-4 <cr>	15	shell	tty1	34	172.26.158.235
2/25/2008	17:40:58	admin2	Group 1	write <cr>	15	shell	tty1	35	172.26.158.235

AAA command accounting is enabled with the following global configuration command:

```
aaa accounting exec
```

A sample configuration using a named method list and AAA server group, plus its enforcement on VTY lines is shown below:

```
!
aaa accounting commands 15 account-exec-list start-stop group tacacs-group
!
line vty 0 4
 accounting commands 15 account-exec-list
!
```

Note When **command accounting** is enabled, *all* commands entered in enable mode will be logged in the accounting records on the accounting host. Consequently, any changes to sensitive information on a device, such as an enable secret, should *not* be entered on the CLI, unless command accounting is temporarily disabled. The recommended approach is to update the configuration offline and securely download the new configuration to the device.

For more information on the **aaa accounting** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_a1.html#wp1038916

For more information on AAA, see AAA Services, page 2-14.

AAA System Accounting

System accounting provides information about system-level events not associated with users, including when the system reboots or when accounting is turned on or off.

In Cisco IOS, AAA system accounting is enabled with the following command:

```
aaa accounting system
```

A sample configuration enabling system accounting for start and stop events to a AAA server group is shown below.

```
Router (config)# aaa accounting system default start-stop group tacacs-group
```

System accounting does not support named accounting but does support named AAA server groups.

A sample system accounting record indicating that AAA accounting has been turned off is shown below.

```
Mon Feb 25 17:13:42 2008        172.26.158.234    unknown unknown unknown start
task_id=25  service=system  event=sys_acct  reason=reconfigure
```

Syslog Login Success and Failure Notifications

Cisco IOS offers the ability to send a syslog trap upon login success or login failure and can be used to complement the AAA EXEC accounting records.

This feature is one of the IOS login enhancements introduced in Cisco IOS Release 12.3(4)T and 12.2(25)S.

The generation of syslog traps for successful and failed login attempts is enabled using the following commands respectively.

```
Router(config)# login on-success log
Router(config)# login on-failure log
```

A sample syslog message for a successful login is shown below:

```
Sep 25 12:49:32.465 UTC: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: admin2
] [Source: 172.26.158.234] [localport: 22] at 12:49:32 UTC Thu Sep 25 2003
```

A sample syslog message for a failed login attempt is shown below:

```
Sep 25 13:19:46.864 UTC: %SEC_LOGIN-4-LOGIN_FAILED: Login failed [user: ] [Sourc
e: 172.26.158.234] [localport: 22] [Reason: Login Authentication Failed] at 13:1
9:46 UTC Thu Sep 25 2003
```

For more information on the **login on-success** and **login on-failure** commands, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_k1.html#wp1031689

Configuration Change Notification and Logging

Cisco IOS offers a Configuration Change Notification and Logging (Config Log Archive) feature which tracks commands executed in configuration mode through the CLI or HTTP. A syslog notification of configuration changes may also be enabled.

The log record includes the following information:

- Configuration command that was executed
- Configuration mode in which the command was executed
- User who executed the command
- Time at which the command was executed
- Configuration change sequence number
- Parser return codes for the command

For more information on the Configuration Change Notification and Logging (Config Log Archive) feature, refer to the following URL:

http://www.cisco.com/en/US/partner/products/ps6350/products_configuration_guide_chapter09186a0080454f73.html

Archive configuration change logging is enabled as shown in the following example:

```
! Enter archive mode
Router(config)# archive
!
! Enter the archive configuration change logger configuration mode
Router(config-archive)# log config
!
! Enable configuration change logging
Router(config-archive-log-config)# logging enable
!
! Set the maximum number of configuration change log entries as 200
Router(config-archive-log-config)# logging size 200
!
! Prevent passwords from being displayed in the configuration log
Router(config-archive-log-config)# hidekeys
!
! Enable configuration change messages to be sent to a syslog server
Router(config-archive-log-config)# notify syslog
!
```

For more information on the **archive** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/fundamentals/command/reference/cf_a1.html#wp1018716

Displaying Configuration Change Log Entries

The configuration change log may be viewed using the command **show archive log config** command as follows:

```
Router# show archive log config all
```

http://www.cisco.com/en/US/partner/docs/ios/12_4/cfg_fund/command/reference/cfn_07h.html#wp1034746

A line-by-line comparison of a specified configuration file to the running configuration file, which generates a list of the configuration lines that do not appear in the running configuration file, can be viewed using the command **show archive config incremental-diffs** command, refer to the following URL:

```
Router# show archive config incremental-diffs nvram:startup-config
```

http://www.cisco.com/en/US/partner/docs/ios/12_4/cfg_fund/command/reference/cfn_07h.html#wp1039115

A line-by-line comparison of any two configuration files (accessible through the Cisco IOS File System [IFS]) and a list of the differences between them can be generated with the **show archive config differences** command as follows:

```
Router# show archive config differences nvram:startup-config
```

For more information on the **show archive log** command, refer to the following URL:

http://www.cisco.com/en/US/partner/products/ps6350/products_configuration_guide_chapter09186a0080454f73.html

For more information on the **show archive config incremental-diffs** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/fundamentals/command/reference/cf_s1.html#wp1077902

For more information on the **show archive config differences** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/fundamentals/command/reference/cf_s1.html#wp1075054

General Device Access and Configuration Change Logging Best Common Practices

General device access and configuration change logging best common practices include:

- Logs should be written to a secure server through a secure, reliable path.
- Ensure that the logs and the servers themselves are properly secured.
- Logged information should be reviewed on a regular basis.
- A high rate of failed authentication attempts may indicate password guessing.
- A high rate of failed authorization attempts may indicate device compromise.
- Only required information should be logged to minimize the amount of traffic generated and the post-processing required.
- It is critical to ensure that NTP is employed to generate logs with consistent, synchronized timestamps across the entire network infrastructure. For more information, refer to Time Synchronization, page 5-2.

File Transfer

Files which may need to be transferred to network infrastructure devices include image and configuration files. The secure transfer of these files is critical to network infrastructure security.

Common file transfer techniques include:

- File Transfer Protocol (FTP)
- Trivial File Transfer Program (TFTP)
- Secure Copy (SCP)

Note Both FTP and TFTP offer minimal security and transfer data in clear text, whereas SCP uses SSH for authentication and encryption. Consequently, SCP should be used wherever feasible.

File Transfer Protocol (FTP)

FTP requires a username and password in order to authenticate access to the FTP server, however all data is sent in clear text. Consequently, though FTP offers a minor advantage over using TFTP to upload a configuration file to a server, it remains vulnerable to sniffing of both the username and password, as well as the configuration file itself. This may be of significant concern if one considers the sensitive data within a configuration file.

It is recommended that Secure Copy (SCP) be used as an alternative to FTP wherever feasible. If FTP must be used, it is recommended that transactions be conducted from a loopback or out-of-band (OOB) interface on a device, thereby enabling access to the FTP server to be restricted to authorized source IP addresses.

```
Router(config)# ip ftp source-interface <Loopback-OOB>
```

Source IP spoofing protection should also be deployed.

It is recommended that the FTP username and password are not defined in the configuration file itself but entered as part of the copy command.

The use of IPSec between a device and the FTP server itself, or perhaps its network subnet, should be considered for transfers over public networks.

Trivial File Transfer Program (TFTP)

TFTP transactions are not authenticated and all data is sent in clear text. Consequently, TFTP is vulnerable to unauthorized access, as well as sniffing. This may be of significant concern if one considers the sensitive data within a configuration file.

It is recommended that Secure Copy (SCP) be used as an alternative to TFTP wherever feasible. If TFTP must be used, it is recommended that transactions be conducted from a loopback or out-of-band (OOB) interface on a device, thereby enabling access to the TFTP server to be restricted to authorized source IP addresses.

```
Router(config)# ip tftp source-interface <Loopback-OOB>
```

Source IP spoofing protection should also be deployed.

The use of IPSec between a device and the TFTP server itself, or perhaps its network subnet, should be considered for transfers over public networks.

Secure Copy (SCP)

SCP relies on SSH for secure authentication and transport, enabling the secure and authenticated copying of files. Consequently, SSH must be already configured on a device in order for SCP to be enabled. See the Secure Shell (SSH) section for full details on how to enable SSH.

In Cisco IOS, SCP can be enabled with the following global configuration command:

```
Router(config)# ip scp server enable
```

A secure copy is initiated through the standard **copy** command. SCP is available as one of the transfer options. For example:

```
cr20-6500-4#copy disk1:remote.cfg scp://admin2@172.26.159.164
Address or name of remote host [172.26.159.164]?
Destination username [admin2]?
Destination filename [remote.cfg]?
Writing remote.cfg
Password:
!!
6258 bytes copied in 7.192 secs (870 bytes/sec)
cr20-6500-4#
!
```

AAA-based authentication and authorization should be employed to ensure only authorized users are able to perform an SCP operation.

```
! AAA authentication and authorization must be configured properly for SCP to work.
aaa new-model
aaa authorization exec default group tacacs-group local
aaa authorization exec default group tacacs-group local
username <admin-user> privilege 15 password <password>
! SSH must be configured and functioning properly.
ip ssh time-out 120
ip ssh authentication-retries 3
```

For more information on the **ip scp server enable** command, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_i2.html#wp1031870

Device Software Image Verification

The software installed and running on a network infrastructure device should be verified to ensure that it is valid. Cisco IOS offers an image verification feature which validates the MD5 digest of an IOS image. If available, digital signing of software should be leveraged to address authentication and non-repudiation issues.

IOS Software Image Verification

The Cisco IOS software image verification feature provides a user-friendly mechanism for validating the MD5 digest of an IOS image. This may be configured to occur automatically, upon any ‘copy’ or ‘reload’, or it may be enabled as a manual option when entering the ‘copy’ or ‘reload’ command. A verification may also be initiated from the CLI.

Image verification is not currently supported on non-IOS image files.

To enable automatic image verification, use the Cisco IOS command:

```
Router(config)# file verify auto
```

Manual image verification may be initiated from the CLI using the `verify` command:

```
Router# verify location://image
```

Manual image verification may be initiated from the CLI during a copy:

```
Router# copy /verify <source> <dest>
```

Manual image verification may also be initiated from the CLI during a reload:

```
Router# reload /verify
```

For more information on the Image Verification feature, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/security/configuration/guide/sec_image_verifctn_ps6350_TSD_Products_Configuration_Guide_Chapter.html

Infrastructure Management Network

An infrastructure management network refers to the network carrying control and management plane traffic (such as NTP, SSH, Telnet, SNMP, syslog, FTP, etc) for the infrastructure devices themselves. Device access can be through the console, as well as through Ethernet or other network management interfaces.

This control and management plane traffic is critical to network operations, providing visibility into and control over the network. Consequently, a well-designed and secure infrastructure management network is critical to the overall security and operation of a network.

One of the key recommendations for a secure infrastructure management network is the separation of management and data traffic in order to ensure remote manageability even under high load and traffic conditions.

Infrastructure management network implementation approaches include:

- Out-of-band (OOB)

  An OOB management network consists of a network which is completely independent and physically disparate from the data network that it helps to manage. This is also sometimes referred to as a Data Communications Network (DCN). Network devices may connect to the OOB network in different ways:

  - Some network devices feature built-in management interfaces that may be used to connect to the OOB network.
  - Some platforms allow the configuration of physical interfaces as dedicated management interfaces. For example, in Cisco IOS Management Plane Protection restricts management traffic to predefined management interfaces.
  - Network devices may also connect to the OOB network with dedicated data interfaces. In this case, ACLs should be deployed to ensure management traffic is only handled by the dedicated interfaces.

- **Pseudo out-of-band**

  A pseudo out-of-band management network uses the same physical infrastructure as the data network but provides logical separation through the virtual separation of traffic, such as using VLANs or VPNs.

- **In-band**

  An in-band management network uses the same physical and logical paths as the data traffic.

An OOB management network provides the maximum visibility and control as it is not impacted by incidents on the data network. An OOB management network does require the deployment and management of separate infrastructure, including, perhaps, redundancy for maximum availability.

However, management network traffic does not typically demand high bandwidth or high performance devices and so the costs are typically reasonable. In addition, the number of devices required is typically small, requiring only sufficient density to provide connectivity to each infrastructure device being managed.

Ultimately, the design decision requires a per-customer analysis of risk versus benefits and costs.

Some general considerations include:

- An isolated OOB management network maximizes visibility and control over the network even during disruptive events.
- Transmitting network telemetry over an OOB network minimizes the chance for disruption of the very information which provides critical network visibility.
- In-band management access to network infrastructure, hosts, etc, is vulnerable to complete loss in the event of a network incident, removing all network visibility and control. Appropriate QoS controls should be put in place to mitigate this.
- Many network infrastructure devices feature interfaces which may be dedicated to device management, including serial console ports and Ethernet management interfaces.
- An OOB management network can typically be deployed at a reasonable cost, since management network traffic does not typically demand high bandwidth, nor high performance, devices and only requires port density sufficient to support connectivity to each infrastructure device.

Device Management Best Common Practices

- Enforce a strong password policy
- Assign and enforce per-user accounts

  Each user should have an individual, unique username and password
- Remove default accounts
- Change default passwords
- Grant minimum access privileges
- Force users to periodically change their passwords
- Selectively use SNMP and treat community strings like root passwords
- Employ secure management protocols where available, including SSH, SCP, SSL, OTP etc.
- If insecure management protocols such as Telnet, syslog, SNMP, TFTP, FTP, etc. are required, consider out- of-band (OOB) management
- If OOB management is not possible, restrict access to the management protocols using the “set ip permit” lists on the management protocols
- Put the management VLAN into a dedicated non-standard VLAN where nothing but management traffic resides and consider physically back-hauling this interface to the management network
- Review the password recovery settings