Best Practices: Server Security Hardening

Revised: Month Day, Year-<required>

This appendix provides background information and advice on the best ways to enhance the security of your Cisco Prime Infrastructure servers.

Hardening Server Security

The following sections explain how to enhance server security by eliminating or controlling individual points of security exposure.

Related Topics

- Disabling Insecure Services
- Disabling Root Access
- Using SNMPv3 Instead of SNMPv2
- Authenticating With External AAA
- Enabling NTP Update Authentication
- Enabling Certificate-Based OCSP Authentication
- Importing Client Certificates Into Web Browsers
- Enabling OCSP Settings on the Prime Infrastructure Server
- Setting Up Local Password Policies
- Checking On Server Security Status

Disabling Insecure Services

You should disable non-secure services if you are not using them. For example: TFTP and FTP are not secure protocols. These services are typically used to transfer firmware or software images to and from network devices and Prime Infrastructure. They are also used for transferring system backups to external storage. We recommend that you use secure protocols (such as SFTP or SCP) for such services.

To disable FTP and TFTP services:

Step 1 Log in to Prime Infrastructure with a user ID with administrator privileges.

Step 2 Select **Administration > Settings > System Settings > General > Server.**

Step 3 Select the **Disable** buttons for **FTP** and **TFTP**.

Step 4 Restart Prime Infrastructure to apply the updated settings.

Disabling Root Access

Administrative users can enable root shell access to the underlying operating system for trouble shooting purposes. This access is intended for Cisco Support teams to debug product-related operational issues. We recommend that you keep this access disabled, and enable it only when required. To disable root access, run the command **root_disable** from the command line (see Connecting Via CLI).

During installation, Prime Infrastructurealso creates a web root user account, prompting the installer for the password to be used for this account. The web root account is needed to enable first-time login to the Prime Infrastructure server and its web user interface. We recommend that you never use this account for normal operations. Instead, use it to create user IDs with appropriate privileges for day-to-day operations and network management, and administrative user IDs for managing Prime Infrastructure itself. Once these user accounts are created, disable the default “web root” account created at install time, and create user accounts using your administrative user IDs thereafter.

If you forget the shell password, you can recover (and then reset) the shell password by following the steps to recover the administrator password. See Recovering Administrator Passwords on Virtual Appliances. Because recovering the administrator password requires the Prime Infrastructure server to reboot, your system might go down for approximately 20 minutes.

To disable the root accounts:

Step 1 Open a CLI session with the Prime Infrastructure server (see Connecting Via CLI). Do not enter “configure terminal” mode.

Step 2 Disable the web root account by entering the following command:

```
PIServer/admin# ncs webroot disable
```

Prime Infrastructure disables the web root account.

Step 3 Disable the root shell account by entering the following command at the prompt:

```
PIServer/admin# shell disable
```

Prime Infrastructure will prompt you for the root shell account password. Enter it to complete disabling of the root shell account.

Using SNMPv3 Instead of SNMPv2

SNMPv3 is a higher-security protocol than SNMPv2. You can enhance the security of communications between your network devices and the Prime Infrastructure server by configuring the managed devices so that management takes place using SNMPv3 instead of SNMPv2.

You can choose to enable SNMPv3 when adding new devices, when importing devices in bulk, or as part of device discovery. See Related Topics for instruction on how to perform each task.

Related Topics

- Using SNMv3 When Adding Devices
- Using SNMv3 When Importing Devices
- Using SNMv3 When Running Discovery

Using SNMv3 When Adding Devices

To specify SNMPv3 when adding a new device:

Step 1 Select **Inventory > Device Management > Network Devices**

Step 2 Choose **Add Device**.

Step 3 In the **SNMP Parameters** area, in **Version**, select **v3**.

Step 4 Complete the other fields as appropriate, then click **Add**.

Related Topics

- Using SNMv3 When Importing Devices
- Using SNMv3 When Running Discovery
- Using SNMPv3 Instead of SNMPv2

Using SNMv3 When Importing Devices

To specify use of SNMPv3 when importing devices in bulk:

Step 1 Select **Inventory > Device Management > Network Devices.**

Step 2 Choose **Bulk Import.** The Bulk Import page appears.

Step 3 Download the device add sample template from the “here” link on the Bulk Import page.

Step 4 Edit the template file using any CSV-compatible application. For each row representing a device in the CSV import file:

a. In the **snmp version** column, enter **3**.

b. Enter appropriate values in the **snmpv3_user_name**, **snmpv3_auth_type**, **snmpv3_auth_password**, **snmpv3_privacy_type**, and **snmpv3_privacy_password** columns.

c. Complete other columns as appropriate for your devices.

Step 5 Select **Inventory > Device Management > Network Devices**, then click **Bulk Import** and import your modified CSV file.

Related Topics

- Using SNMv3 When Adding Devices
- Using SNMv3 When Running Discovery
- Using SNMPv3 Instead of SNMPv2

Using SNMv3 When Running Discovery

To specify SNMPv3 as part of device discovery:

Step 1 Select **Inventory > Device Management > Discovery**. The Discovery Jobs page appears.

Step 2 Click the **Discovery Settings** link in the upper right corner of the page. The Discovery Settings page appears.

Step 3 Choose **New** to add new SNMP v3 credentials.

Step 4 Complete the fields as needed.

Step 5 Click **Save** to save the SNMPv3 settings and use them thereafter.

Related Topics

- Using SNMv3 When Adding Devices
- Using SNMv3 When Importing Devices
- Using SNMPv3 Instead of SNMPv2

Authenticating With External AAA

User accounts and password are managed more securely when they are managed centrally, by a dedicated, remote authentication server running a secure authentication protocol such as RADIUS or TACACS+.

You can configure Prime Infrastructure to authenticate users using external AAA servers. You will need to access the **Administration > Users > Users, Roles & AAA** page to set up external authentication via the Prime Infrastructure graphic user interface (GUI). You can also set up external authentication via the command line interface (CLI). See Related Topics for instructions on how to set up AAA using each method.

Related Topics

- Setting Up External AAA Via GUI
- Setting Up External AAA Via CLI

Setting Up External AAA Via GUI

To set up remote user authentication via the GUI:

Step 1 Log in to Prime Infrastructure with a user ID that has administrator privileges.

Step 2 Select **Administration > Users > Users, Roles & AAA > TACACS+** or **Administration > Users > Users, Roles & AAA > RADIUS.**

Step 3 Enter the TACACS+ or RADIUS server IP address and shared secret in the appropriate fields.

Step 4 Select **Administration > Users > Users, Roles & AAA > AAA Mode Settings.**

Step 5 Set the AAA mode as appropriate.

Related Topics

- Authenticating With External AAA
- Setting Up External AAA Via CLI

Setting Up External AAA Via CLI

To set up remote user authentication via the CLI:

Step 1 Log in to Prime Infrastructure using the command line, as explained in Connecting Via CLI. Be sure to enter “configure terminal” mode.

Step 2 At the prompt, enter the following command to setup an external TACACS+ server:

`PIServer/admin/terminal#` **aaa authentication tacacs+ server *tacacs-ip* key plain *shared-secret***

Where:

- ***tacacs-ip*** is the IP address of an active TACACS+ server.
- ***shared-secret*** is the plain-text shared secret for the active TACACS+ server.

Step 3 At the prompt, enter the following command to create a user with administrative authority, who will be authenticated by the above AAA server:

`PIServer/admin/terminal#` **username *username password* remote role admin email *emailID***

Where:

- ***username*** is the name of the user ID.
- ***password*** is the plain-text password for the user.
- ***emailID*** is the email address of the user (optional).

Related Topics

- Authenticating With External AAA
- Setting Up External AAA Via GUI

Enabling NTP Update Authentication

Network Time Protocol (NTP) version 4, which authenticates server date and time updates, is an important way to harden server security. Note that you can configure a maximum of three NTP servers with Prime Infrastructure.

To set up authenticated NTP updates:

Step 1 Log in to Prime Infrastructure using the command line, as explained in Connecting Via CLI.Be sure to enter “configure terminal” mode.

Step 2 At the prompt, enter the following command to setup an external NTPv4 server:

PIServer/admin/terminal**# ntp server *serverIP userID* plain *password***

Where:

- ***serverIP*** is the IP address of the authenticating NTPv4 server you want to use.
- ***userID*** is the md5 key id of the NTPv4 server.
- ***password*** is the corresponding plain-text md5 password for the NTPv4 server.

For example: `ntp server 10.81.254.131 20 plain MyPassword`

Step 3 To ensure that NTP authentication is working correctly, test it by executing the following commands:

- To check the NTP update details: **sh run**
- To check NTP sync details: **sh ntp**

Enabling Certificate-Based OCSP Authentication

You can further enhance the security of Prime Infrastructure's interaction with its web clients by setting up certificate-based client authentication using the Online Certificate Status Protocol (OCSP).

With this form of authentication, Prime Infrastructure validates the web client's certificate and its revocation status before permitting the user to access the login page. Checking the revocation status ensures that the issuing Certificate Authority (CA) has not already revoked the certificate.

Prime Infrastructure uses OCSP to check the certificate's revocation status. OCSP is a real-time certificate status check mechanism, which is faster and more reliable than other methods. New, internet standard protocol, not proprietary, most browsers support it, widely accepted, and DOD-compliant.

Before You Begin

You will want to ensure that:

- Prime Infrastructure is configured to authorize user access via an external AAA server using a secure protocol, such as RADIUS or TACACS+. The US Department of Defense and other security agencies recommend doing so as a way to ensure secure authentication. See "Authenticating With External AAA" in Related Topics for more information. This permits two-factor authentication: certificate authentication takes place separately from user ID and password authentication.
- You have set up a repository to store certificates. There are very few restrictions on how you do this. The repository can be located on storage media local to the Prime Infrastructure server or on a remote host. If it is remote, it can be located on a dedicated server you set up or on a shared certificate server used throughout your organization. Be sure that any remote repository you use is accessible from the Prime Infrastructure server via a supported protocol (NFS, FTP, or SFTP).
- You know the name of the certificate repository, the name of the folder within the repository where certificate files are stored, and the name and password of a user with read/write access to that repository and folder.
- The certificate files exist in the certificate repository.
- Your organization's DNS servers are able to resolve the URLs of the OCSP responders maintained by the CA who issued your certificates. These OCSP responder URLs will be embedded in the certificate files (such as "OCSP.Responder.Service"). IP addresses are not embedded in the certificate files, so these URLs must be resolvable by DNS.

After You Finish

Once you have enabled this form of authentication, every client web browser used to access Prime Infrastructure must import the client certificates. See "Importing OCSP-Verified Certificates Into Web Clients" in Related Topics for more information.

Step 1 Log in to Prime Infrastructure using the command line, as explained in "Connecting Via CLI" in Related Topics. Be sure to enter "configure terminal" mode.

Step 2 Run the following commands in the order given to create an alias for the certificate repository and configure Prime Infrastructure to access that alias:

```
PIServer/admin(config)# repository CertRepoName
PIServer/admin(config-repository) # url proto://CAPath
PIServer/admin(config-repository)# user username password type pword
```

Where:

- ***CertRepoName*** is the name of the certificate repository (for example: **MyCertRepo**)

- ***proto*** is the name of the protocol used to access the repository (that is: **NFS**, **FTP**, or **SFTP**).
- ***CAPath*** is the complete URL and path to the location where the certificates are stored.
- ***username*** is the name of the user who will be accessing the certificates in the repository. This must be an existing user already given permission to access ***CAPath***.
- ***type*** is the password encryption type (either **plain** for plain text, or **hash** for an encrypted password).
- ***pword*** is the corresponding password for the user specified in ***username*** (in plain text or encrypted form, depending on the value of ***type).***

Step 3 Run the following commands to verify that the certificates are available at the path on the certificate repository:

`PIServer/admin(config-repository)#` **exit**

`PIServer/admin(config)#` **exit**

`PIServer/admin#` **show repository *CertRepoName***

The last command will return a list of the certificates stored in the repository. The certificates you want should be in the list. For example, if you have more than one certificate file, you might see a listing like this:

```
certnew_latest.cer
certnew_sub_ca1.cer
```

Step 4 Run the following command to install the certificates into the Prime Infrastructure keystore repository, creating an alias for each file (if you have more than one certificate file, you will need to run this comamnd more than once):

`PIServer/admin#` **ncs key importcert *CertAlias CertFile* repository *CertRepoName***

Where:

- ***CertAlias*** is the alias you want to assign to the certificate file. The alias must be unique for each file.
- ***CertFile*** is the file name of the certificate file stored in ***CertRepoName***.

For example: To continue using the sample certificate file names given in Step 3, you might execute the following commands:

`PIServer/admin#` **ncs key importcacert OCSP-CA-CERT certnew_latest.cer repository MyCertRepo**

`PIServer/admin#` **ncs key importcacert OCSP-SUB-CA-CERT certnew_sub_ca1.cer repository MyCertRepo**

Step 5 After installing the certificates, restart the Prime Infrastructure server as explained in “Restarting Prime Infrastructure”.

Step 6 Once the server is restarted: Log in to Prime Infrastructure via the command line as you did in Step 1 and display the list of certificates installed in the Prime Infrastructure keystore:

`PIServer/admin#` **ncs key listcacerts**

The list of installed certificates should contain the certificates you imported in Step 4.

Step 7 Run the following command to enable client certificate authentication on Prime Infrastructure:

`PIServer/admin#` **ncs run client-auth enable**

Step 8 After enabling client certificate authentication: Restart the Prime Infrastructure server again, as explained in “Restarting Prime Infrastructure”.

Related Topics

- Authenticating With External AAA
- Connecting Via CLI
- Using Remote Backup Repositories
- Restarting Prime Infrastructure
- Importing Client Certificates Into Web Browsers

Importing Client Certificates Into Web Browsers

Users accessing Prime Infrastructure servers with certificate authentication must import client certificates into their browsers in order to authenticate. Although the process is similar across browsers, the actual details vary with the browser. The following procedure assumes that your users are using a Prime Infrastructure compatible version of Firefox.

Before You Begin

You must ensure that the user importing the client certificates has:

- Downloaded a copy of the certificate files to a local storage resource on the client machine
- If the certificate file is encrypted: The password with which the certificate files were encrypted.

Step 1 Launch Firefox and enter the following URL in the location bar: **about:preferences#advanced**.

Firefox displays its **Options** > **Advanced** tab.

Step 2 Select **Certificates** > **View Certificates** > **Your Certificates**, then click **Import...**.

Step 3 Navigate to the downloaded certificate files, select them, then click **OK** or **Open**.

Step 4 If the certificate files are encrypted: You will be prompted for the password used to encrypt the certificate file. Enter it and click **OK**.

The certificate is now installed in the browser.

Step 5 Press **Ctrl+Shift+Del** to clear the browser cache,

Step 6 Point the browser to the Prime Infrastructure server using certificate authentication.

You will be prompted to select the certificate with which to respond to the server authentication requested. Select the appropriate certificate and click **OK**.

Related Topics

- Authenticating With External AAA
- Using Remote Backup Repositories
- Enabling Certificate-Based OCSP Authentication

Setting Up SSL Certification

The Secure Sockets Layer (SSL) Certification is used to ensure secure transactions between a web server and the browsers. Installing the certificates allows your web browser to trust the identity and provide secure communications which are authenticated by a certificate signing authority (CSA).

These certificates are used to validate the identity of the server or website and are used to generate the encryption key used in the SSL. This encryption protects the information being passed between the server and the client.

Related Topics

- Setting Up SSL Client Certification
- Setting Up SSL Server Certification

Setting Up SSL Client Certification

To set up the SSL *client* certificate authentication, follow the steps below. These steps use the US Department of Defense (DoD) as an example of a Certificate Signing Authority (CSA), but you may use any CSA that authenticates SSL certificates.

Note that access to the keytool utility, available in JDK, is required in this method of creating SSL certificates. Keytool is a command-line tool used to manage keystores and the certificates.

Step 1 Create SSL Client Certificate using the below command.

```
% keytool -genkey -keystore nmsclientkeystore -storetype pkcs12 -keyalg RSA -keysize 2048
-alias nmsclient -dname "CN=nmsclient, OU=WNBU, O=Cisco, L=San Jose, ST=CA, C=US"
-storepass nmskeystore
```

Provide the Key Algorithm as RSA and KeySize as 1024 or 2048.

Step 2 Generate the Certificate Signing Request (CSR) using the below command.

```
% keytool -certreq -keyalg RSA -keysize 2048  -alias nmsclient -keystore nmsclientkeystore
-storetype pkcs12 -file <csrfilename>
```

Provide the Key Algorithm as RSA and KeySize as 1024 or 2048 and provide a certificate file name.

Step 3 Send the generated CSR file to DoD. The DoD issues the corresponding signed certificates.

The CSR reply is through dod.p7b file. In addition you should also receive the root CA certificates.

Please makes sure to retrieve the PKCS7 encoded certificates; Certificate Authorities provide an option to get the PKCS7 encoded certificates.

Step 4 Import the CSR reply into the Keystore using the command:

```
% keytool -import dod.p7b -keystore nmsclientkeystore -storetype pkcs12
-storepass nmskeystore
```

Step 5 Check the formats of root CA certificates received. They must be base-64 encoded. If they are not base-64 encoded, use the OpenSSL command to convert them to this format.

```
% openssl x509 -in rootCA.cer -inform DER -outform PEM -outfile rootCA.crt
% openssl x509 -in DoD-sub.cer -inform DER -outform PEM -outfile rootCA.crt
```

Convert both root CA certificate and sub-ordinate certificates received.

In case you received both root CA certificate and the sub-ordinate certificate, you have to bundle them together using the below command:

```
% cat DoD-sub.crt > ca-bundle.crt
% cat DoD-rootCA.crt >> ca-bundle.crt
```

Step 6 To set up SSL Client Authentication using these certificates, enable SSL Client Authentication in Apache in the **ssl.conf** file located in <NCS_Home>/webnms/apache/ssl/backup/ folder.

```
SSLCACertificationPath conf/ssl.crt
SSLCACertificationFile conf/ssl.crt/ca-bundle.crt
SSLVerifyClient  require
SSLVerifyDepth 2
```

SSLVerifyDepth depends on the level of Certificate Chain. In case you have only 1 root CA certificate, this should be set to 1. In case you have a certificate chain (root CA and subordinate CA), this should be set to 2.

Step 7 Install the DoD root CA certificates in Prime Infrastructure.

Step 8 Import the nmsclientkeystore in your browser.

Related Topics

- Setting Up SSL Certification
- Setting Up SSL Server Certification

Setting Up SSL Server Certification

Step 1 Generate the Certificate Signing Request (CSR).

```
% ncs key genkey -csr <csrfilename> repository <repositoryname>
```

Step 2 Import the Signed Certificate using the below command:

```
% ncs key importcacert <aliasname> <ca-cert-filename> repository <repositoryname>
```

Prime Infrastructure stores the self-signed certificate at /opt/CSCOncs/httpd/conf/ssl.crt. The imported certificates/keys are stored at /opt/CSCOncs/migrate/restore.

Related Topics

- Setting Up SSL Certification
- Setting Up SSL Client Certification

Enabling OCSP Settings on the Prime Infrastructure Server

Online Certificate Status Protocol (OCSP) enables certificate-based authentication for web clients using OCSP responders. Typically, the OCSP responder's URL is read from the certificate's Authority Information Access (AIA). As a failover mechanism, you can configure the same URL on the Prime Infrastructure server as well

To set up a custom URL of an OCSP responder, follow the steps below.

Step 1 Log in to the Prime Infrastructure server using the command line, as explained in Connecting Via CLI. Do not enter "configure terminal" mode.

Step 2 At the prompt, enter the following command to enable client certificate authentication:

`PIServer/admin#` **ocsp responder custom enable**

Step 3 At the prompt, enter the following command to set the custom OCSP responder URL:

`PIServer/admin#` **ocsp responder set url *Responder#URL***

Where:

- ***Responder#*** is the number of the OCSP responder you want to define (e.g., 1 or 2).
- ***URL*** is the URL of the OCSP responder, as taken from the client CA certificate.

Note that there should be no space between the ***Responder#*** and ***URL*** values**.**

Step 4 To delete an existing custom OCSP responder defined on the Prime Infrastructure server, use the following command:

`PIServer/admin#` **ocsp responder clear url** ***Responder#***

If you do not already know the number of the OCSP responder you want to delete, use the **show security-status** command to view the OCSP responders currently configured on the server. For details, see Checking On Server Security Status.

Setting Up Local Password Policies

If you are authenticating users locally, using Prime Infrastructure's own internal authentication, you can enhance your system's security by enforcing rules for strong password selection.

Note that these policies affect only the passwords for local Prime Infrastructure user IDs. If you are authenticating Prime Infrastructure users via a centralized or remote AAA server, you can enforce similar protections using the functions of the AAA server.

To enforce local password policies:

Step 1 Log in to Prime Infrastructure with a user ID that has administrator privileges.

Step 2 Select **Administration > Users > Users, Roles & AAA > Local Password Policy.**

Step 3 Select the check boxes next to the password policies you want to enforce, including:

- The minimum number of characters passwords must contain.
- No use of the username or "cisco" as a password (or common permutations of these).
- No use of "public" in root passwords.
- No more than three consecutive repetitions of any password character.
- Passwords must contain at least one character from three of the following character classes: upper case, lower case, digit, and special character.
- Whether the password must contain only ASCII characters.
- Minimum elapsed number of days before a password can be reused.
- Password expiration period.
- Advance warnings for password expirations.

If you enable any of the following password policies, you can also specify:

- The minimum password length, in number of characters.
- The minimum elapsed time between password re-uses.
- The password expiry period.
- The number of days in advance to start warning users about future password expiration.

Step 4 Click **Save**.

Disabling Individual TCP/UDP Ports

The following table lists the TCP and UDP ports Prime Infrastructure uses, the names of the services communicating over these ports, and the product's purpose in using them. The "Safe" column indicates whether you can disable a port and service without affecting Prime Infrastructure's functionality.

Table B-1 Prime Infrastructure TCP/UDP Ports

Port	Service Name	Purpose	Safe?
21/tcp	FTP	File transfer between devices and server	Y
22/tcp	SSHD	Used by SCP, SFTP, and SSH connections to and from the system	N
69/udp	TFTP	File transfer between devices and the server	Y
162/udp	SNMP-TRAP	To receive SNMP Traps	N
443/tcp	HTTPS	Primary Web Interface to the product	N
514/udp	SYSLOG	To receive Syslog messages	N
1522/tcp	Oracle	Oracle/JDBC Database connections: These include both internal server connections and for connections with the High Availability peer server.	N
8082/tcp	HTTPS	Health Monitoring	N
8087/tcp	HTTPS	Software updates on HA Secondary Systems	N
9991/udp	NETFLOW	To receive Netflow streams (enabled if Assurance license installed)	N
61617/tcp	JMS (over SSL)	For interaction with remote Plug&Play Gateway server	Y

Checking On Server Security Status

Prime Infrastructure administrators can connect to the server via CLI and use the **show security-status** command to display the server's currently open TCP/UDP ports, the status of other services the system is using, and other security-related configuration information. For example:

Step 1 Log in to Prime Infrastructure using the command line, as explained in Connecting Via CLI. Do not enter "configure terminal" mode.

Step 2 Enter the following command at the prompt:

```
PIServer/admin# show security-status
```

Depending on your settings, you will see output like the following:

```
Open TCP Ports: 21 22 80 443 1522 8082 9992 11011:11014 61617
Open UDP Ports: 69 162 514 9991
FIPS Mode: disabled
TFTP Service: enabled
FTP Service: enabled
JMS port (61617): enabled
Root Access: disabled
Client Auth: enabled
OCSP Responder1: http://10.77.167.65/ocsp
OCSP Responder2: http://10.104.178.99/ocsp
```