Oracle® Communications

Policy Management

Platform Configuration User's Guide

Release 12.5

E94324-01

December 2018

ORACLE

Policy Management Platform Configuration User's Guide, Release 12.5

E94324-01

Contents

3 Synchronizing Files

4 Backing Up and Restoring the System and Server

About This Guide

This chapter describes the organization of the document and provides other information that could be useful to the reader.

How This Guide is Organized

The information in this guide is presented in the following order:

- About This Guide contains general information about this guide, the organization of this guide, and how to get technical assistance.
- Introduction describes how to access the platcfg utility, how to use the utility interface in a Policy Management environment, and troubleshooting.
- Performing Initial Server Configuration describes how to access the platcfg utility and configure your application's initial configuration, and then how to verify the configuration.
- Managing Certificates describes how to access the platcfg utility to manage SSL security certificates, which allow two systems to interact with a high level of security.
- Synchronizing Files describes how and when to synchronize files in clusters.
- Editing Network Interface Ethernet Parameters describes how to manually configure Ethtool options.
- Backing Up and Restoring the System and Server describes how to perform system and server backups and restores.
- Glossary

Intended Audience

This guide is intended for service personnel who are responsible for operating Policy Management systems.

Related Publications

For information about additional publications related to this document, refer to the Oracle Help Center site. See Locate Product Documentation on the Oracle Help Center Site for more information on related product publications.

Locate Product Documentation on the Oracle Help Center Site

Oracle Communications customer documentation is available on the web at the Oracle Help Center (OHC) site, http://docs.oracle.com. You do not have to register to access these documents. Viewing these files requires Adobe Acrobat Reader, which can be downloaded at http://www.adobe.com.

1. Access the Oracle Help Center site at http://docs.oracle.com.
2. Click `Industries`.
3. Under the Oracle Communications subheading, click the `Oracle Communications documentation` link.

   The Communications Documentation page appears. Most products covered by these documentation sets will appear under the headings "Network Session Delivery and Control Infrastructure" or "Platforms."
4. Click on your Product and then the Release Number.

   A list of the entire documentation set for the selected product and release appears.
5. To download a file to your location, right-click the `PDF` link, select `Save target as` (or similar command based on your browser), and save to a local folder.

Customer Training

Oracle University offers training for service providers and enterprises. Visit our web site to view, and register for, Oracle Communications training:

http://education.oracle.com/communication

To obtain contact phone numbers for countries or regions, visit the Oracle University Education web site:

www.oracle.com/education/contacts

My Oracle Support

My Oracle Support is your initial point of contact for all product support and training needs. A representative at Customer Care Center can assist you with My Oracle Support registration.

Call the My Oracle Support main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at http://www.oracle.com/us/support/contact/index.html. When calling, make the selections in the sequence shown below on the Support telephone menu:

1. Select 2 for New Service Request
2. Select 3 for Hardware, Networking and Solaris Operating System Support
3. Select one of the following options:
   - For Technical issues such as creating a new Service Request (SR), Select 1
   - For Non-technical issues such as registration or assistance with MOS, Select 2

You are connected to a live agent who can assist you with My Oracle Support registration and opening a support ticket.

My Oracle Support is available 24 hours a day, 7 days a week, 365 days a year.

Emergency Response

In the event of a critical service situation, emergency response is offered by the Customer Access Support main number at 1-800-223-1711 (toll-free in the US), or by calling the Oracle Support hotline for your local country from the list at http://www.oracle.com/us/support/contact/index.html. The emergency response provides immediate coverage, automatic escalation, and other features to ensure that the critical situation is resolved as rapidly as possible.

A critical situation is defined as a problem with the installed equipment that severely affects service, traffic, or maintenance capabilities, and requires immediate corrective action. Critical situations affect service and/or system operation resulting in one or several of these situations:

- A total system failure that results in loss of all transaction processing capability
- Significant reduction in system capacity or traffic handling capability
- Loss of the system's ability to perform automatic system reconfiguration
- Inability to restart a processor or the system
- Corruption of system databases that requires service affecting corrective actions
- Loss of access for maintenance or recovery operations
- Loss of the system ability to provide any required critical or major trouble notification

Any other problem severely affecting service, capacity/traffic, billing, and maintenance capabilities may be defined as critical by prior discussion and agreement with Oracle.

List of Figures

1

Introduction

This chapter describes how to use the **Oracle Communications Policy Management Platform Configuration** utility (**platcfg**) to configure **Oracle Communications Policy Management** (**Policy Management**) on Policy Management Configuration Management Platform (**CMP**) servers and Policy Management servers.

The reference to Policy Management servers will be used throughout this document to mean the **Policy Management Multimedia Policy Engine** (**MPE**) device, **Policy Management Multi-Protocol Routing Agent** (**MRA**) device, **Policy Management Bandwidth on Demand** (**BoD**) server, **Policy Management Message Distribution Function** (**MDF**) server, and **Policy Management Management Agent** (**MA**) server, collectively. Each server is described individually in detail in their respective manuals.

The pages, tabs, fields, menu items, and functions that you see in the utility depend on your configuration, application, or mode.

platcfg Overview

The `platcfg` utility is a Command Line Interface (CLI) tool that simplifies the execution of tasks that cannot be included in the application software. These tasks include those that affect operating system operations or platform services that are invisible to an application or that are not accessible from the application management controls.

The `platcfg` utility simplifies task execution and reduces the chance of user errors through the use of wizard-like menu options and forms.

You access `platcfg` menus by logging in from a console or logging in remotely. The `platcfg` security actions are centralized at the active CMP server, with all functions propagated automatically to all connected servers.

Accessing platcfg

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

Using Keyboard Actions in the platcfg Utility

Use the following keyboard actions to move and enter information within the platcfg utility:

- Up and down arrows—Moves the action up or down.
- Left and right arrows—Moves the action sideways.
- Enter key—Enters the selected item and moves to the next menu or feature screen.
- First letter—Select the first letter of a menu item to move to that item.

Using the Save Platform Debug Logs Menu to Troubleshoot

Use **Save Platform Debug Logs** to help troubleshoot If a system failure occurs.

Note: The CMP Save Log function includes CMP Audit logs for the previous two months.

Saving Platform Debug Logs

The **Save Platform Debug Logs** menu is used to help you troubleshoot a system failure. You can adjust two settings to limit the size of the saved log files.

Information saved in the logs includes the current state of all logs, all the configuration files, all the system procedure entries, and several miscellaneous files. Output from this process is a single tar/gzip file.

To use the menu:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.

Note: The NetBackup Configuration menu selection only opens if the server is a CMP server.

3. Select **Save Platform Debug Logs** from the Policy Configuration Menu screen and press **Enter**.

4. In the screen that opens, enter values for the following fields:

   - **Record limit for qptrace**

     This field specifies the maximum number of qptrace messages to save. Do not change this setting when generating a save log to debug a problem; only reduce the default number messages when instructed to do so by Customer Support.

   - **Record limit for AppEventLog**

     This field specifies the maximum number of AppEventLog records to save. Do not change this setting when generating a save log to debug a problem; only reduce the default number records when instructed to do so by Customer Support.

   - **Remember count limit settings**

     This field specifies whether or not to retain limit setting from previous log.

   - **Include trace/subact/sync log**

     This field indicates whether to include the extra trace/subact/sync debug records.

   - **Save as**

     This field lists the path and filename of the file being saved.

   Note: **Include trace/subact/sync log** should be left set to **No** unless directed to be set to **Yes** by My Oracle Support.

5. Select **OK** and press **Enter** to save variable changes and generate the tar/gzip file.

   The file is generated and saved in the location you specified.

2

Managing Certificates

This chapter describes how to use the platcfg utility to manage secure sockets layer (SSL) security certificates, which allow systems to interact with a high level of security.

About Security Certificates

To establish a secure (HTTPS) connection between servers in the Policy Management network, or to establish secure connections with third-party systems, you need to create and exchange secure sockets layer (SSL) security certificates, which allow for encrypted communication, before putting the system into production. The platcfg utility supports two types of security certificates: self-signed and third-party.

- Self-signed certificates are created locally on each server using the platcfg utility, then synchronized throughout the Policy Management network to allow encrypted communications between servers. A connection is established between the active servers of a cluster. Because any server in a cluster may become the active server, certificates must be exchanged between all servers in all clusters. To function correctly, the certificates must be current and valid. Self-signed certificates are inherently less secure than third-party signed certificates, so they are not recommended for use in a production environment. Additionally, some external systems may not allow the use of self-signed certificates, which may necessitate the use of third-party certificates.
- Third-party signed certificates are created by an external signing authority. Third-party signed certificates are generated in response to a **Certificate Signature Request** (CSR), which you create locally using the platcfg utility and then send to the third-party signing authority. You then combine it with a current and valid self-signed certificate and synchronize it throughout the Policy Management network.

The following terms relate to the management of certificates:

Certificate
Used by SSL to verify a trusted server; sometimes referred to in platcfg as a Key.

CN (Common Name)
The primary ID inside of a certificate. The Keystore Input Parameters page refers to the CN as **First and Last Name**.

First and Last Name
The primary ID inside of a certificate, also known as the CN.

Key
Another name sometimes used in platcfg to refer to a Certificate.

Local keystore
A file, protected by password-based encryption, that stores self-signed certificates generated on the local servers of a cluster. All servers in a cluster share the same local keystore.

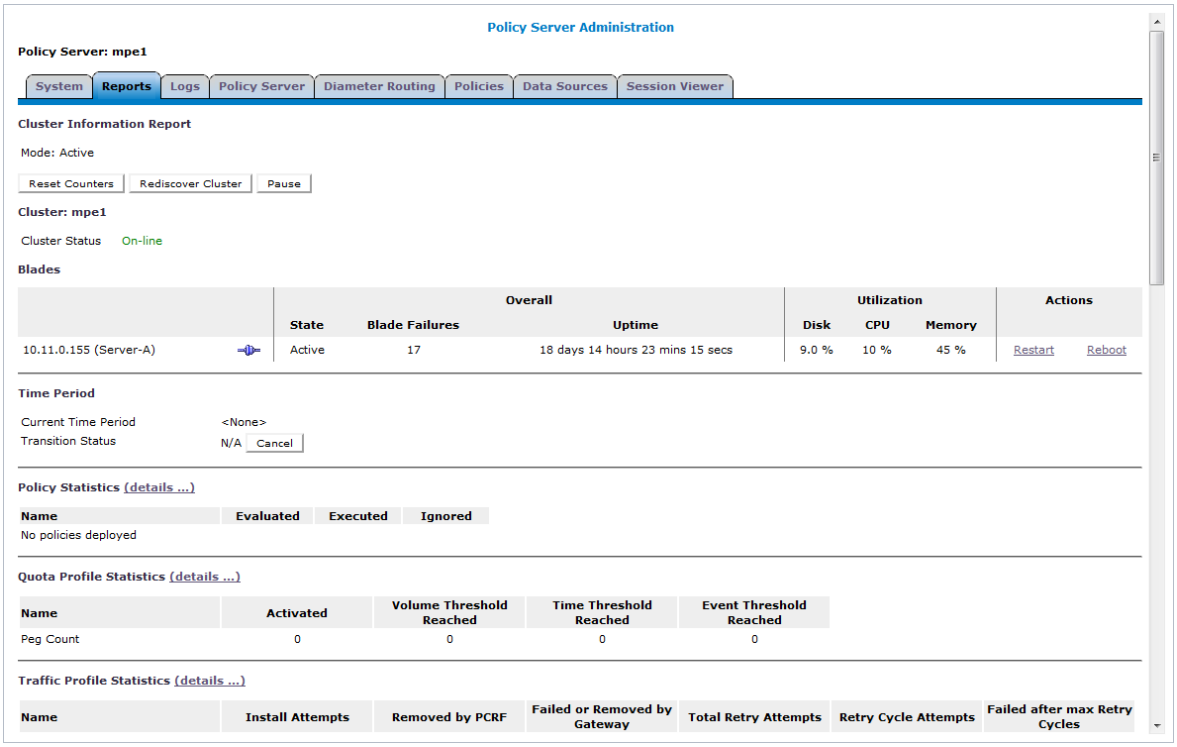
Certificate keystore
A file, protected by password-based encryption, that stores imported certificates generated on other clusters.

When a secure connection is established between the CMP system and a Policy Management cluster:

- An HTTPS session is established and displayed in the URL
- The **System** tab for the cluster displays **Yes** in the **Secure Connection** field
- The **Reports** tab for the cluster displays statistics

Figure 2-1 shows an example of statistics information displayed on the **Reports** tab of the Policy Server Administration page over a secure connection for an MPE cluster.

Figure 2-1 Statistics Displayed Over a Secure Connection



Managing SSL Security Certificates

This section describes how to create and verify self-signed certificates for secure communication between servers and systems.

Creating a Self-signed Certificate

A certificate is used by SSL to verify a trusted server. Certificate creation is performed on the active server in each cluster in the topology and then shared with the other servers of each cluster. This local certificate acts as a Private certificate for the local

server and enables encrypted information to be transferred through a secure connection.

Note: Common Name (CN) is the primary ID inside of a certificate. The Keystore Input Parameters page refers to the CN as **First and Last Name**.

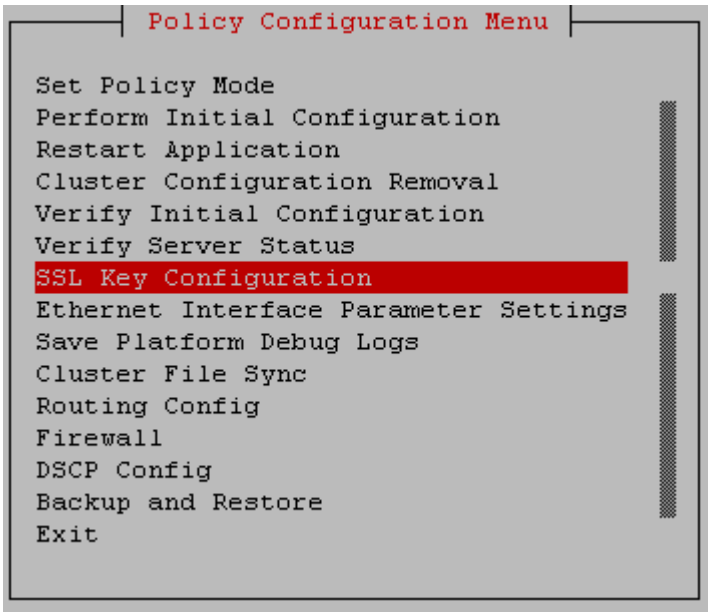
To create a self-signed certificate for a cluster and then synchronize it across the cluster:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

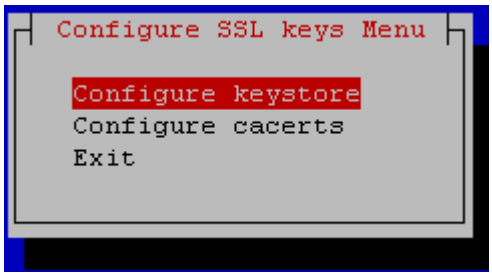
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **SSL Key Configuration** from the Policy Configuration Menu screen and press **Enter**.

Figure 2-2 Policy Configuration Menu—SSL Key Configuration



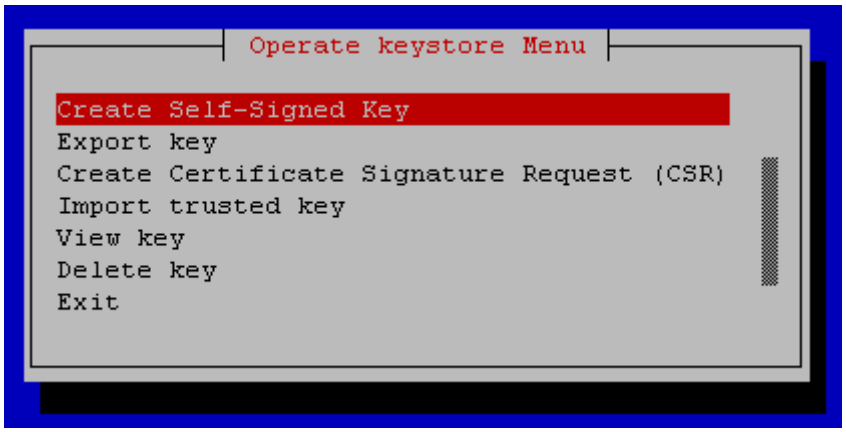
4. Select **Configure keystore** from the Configure SSL keys Menu screen and press **Enter**.

Figure 2-3 Configure SSL keys Menu—Configure keystore



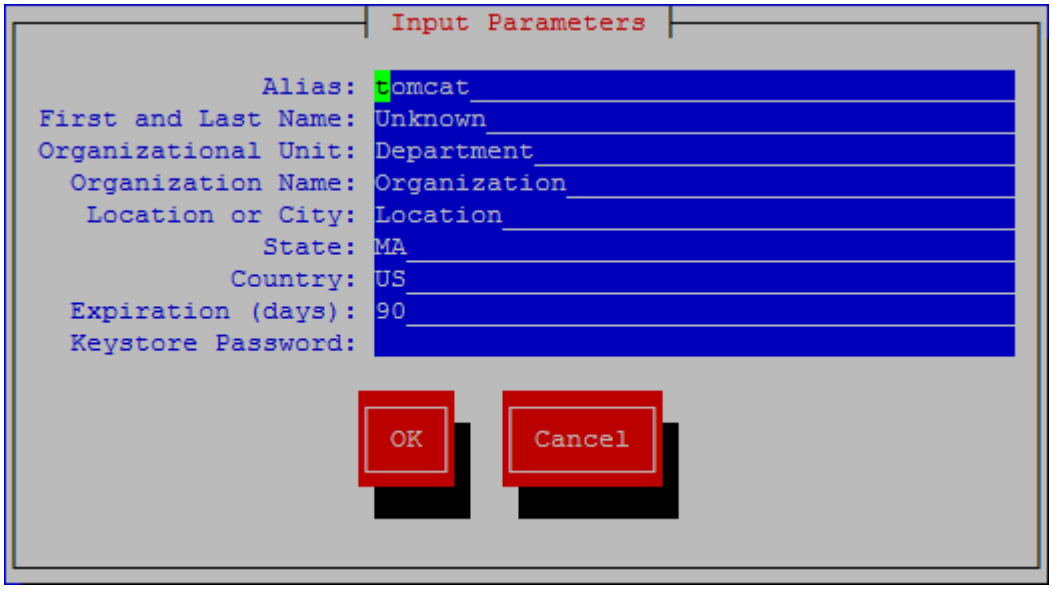
5. Select **Create Self-Signed Key** from the Operate keystore Menu and press **Enter**.

Figure 2-4 Operate keystore Menu



6. Enter information on the Input Parameters screen.

Figure 2-5 Input Parameters



Note: For the **Alias** field, enter `tomcat`.

Note: For the **First and Last Name** field (the CN value), create a unique cluster ID name.

Note: The **Keystore Password** is **changeit**

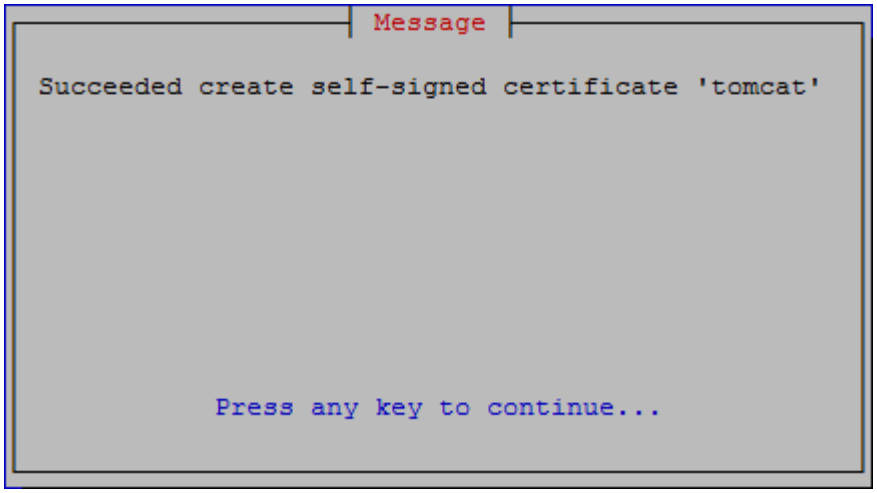
7. When finished entering values, select **OK** and press **Enter**.
8. If there is an existing certificate with the same **Alias** name, the following screen opens:

Figure 2-6 Delete existing certificate

Select **Yes** to remove the old certificate and replace it with a new one with the same name.

9. The following screen opens when the SSL creation is successful.

Figure 2-7 Message



Press **Enter** to return to the previous screen.

10. Select **Cluster File Sync** from the Policy Configuration Menu screen and press **Enter**.

    The self-signed certificate is synchronized to the others servers of the cluster.

11. Select **Restart Application** from the Policy Configuration Menu screen and press **Enter**.

The Policy Management application (the qp_procmgr process) on the active server restarts.

Repeat this procedure for every cluster in the Policy Management network.

Verifying a Self-signed Certificate

After an SSL certificate has been created, verify its attributes before attempting to import or export the certificate to create your secure connection. If the certificate on the host is not the same after it is imported into its peer, the secure connection will not be allowed.

To verify a self-signed certificate:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **SSL Key Configuration** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Configure keystore** from the Configure SSL keys Menu screen and press **Enter**.
5. Select **View key** and press **Enter**.
6. Enter the password, select **OK**, and press **Enter**.
7. Select the certificate and press **Enter**.

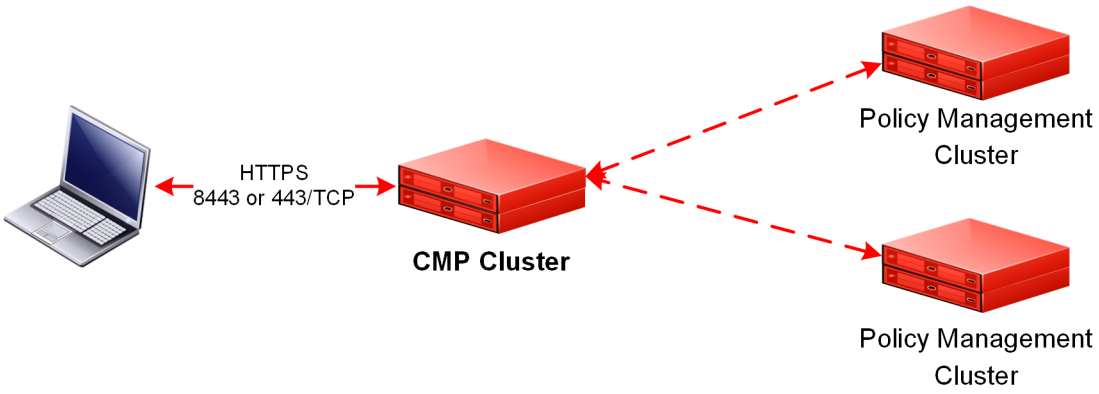
   The certificate opens.
8. Verify the certificate information in the Verify Self-Signed Certificate screen.

   The most important portions of the certificate are the **Alias name**, **Owner**, and **Issuer**. These settings are exported and imported to the other server to establish the secure **HTTP** session.
9. Select **Exit** and press **Enter**.

Using a Local Certificate to Establish a Secure HTTP Web-Browser Session

An HTTPS connection is created between an end user (web browser) and the CMP system by passing a predefined certificate to the end user.

Figure 2-8 Establishing a Secure Session



Note: Web browsers function differently based on their configuration. Review your browser settings before using SSL certificates.

Note: For more information, refer to Creating a Self-signed Certificate and Configuring Firewall Settings.

To force end users to establish an HTTPS session with the CMP system:

1. Exchange and import SSL certificates between the CMP server and the workstation.
2. Enable the firewall on the CMP server.
3. Enable **prefer custom**.
4. Create two customized firewall rules (one for port 80 and one for port 8080) where the allowed host is 0.0.0.0/32.

Note: Because the ports 80 and 8080 conflict with the factory rule that allows anyone access to these ports, using the **prefer custom** option discards this rule, and instead uses the custom rule which allows only 0.0.0.0 to connect via 80 or 8080, which locks down the unencrypted HTTP ports.

About Establishing a Secure Connection Between a CMP System and a Policy Management Server

To establish a secure connection between a CMP system and a Policy Management server, both the CMP system and the Policy Management server must exchange certificates.

Figure 2-9 Exchanging Certificates

The figure shows how the SSL certificate is shared between the clusters. The following certificate exchange is done:

1. The CMP system creates a local certificate and exports the certificate to the Policy Management server.
2. The Policy Management server imports the peer certificate (local certificate created by the CMP system) into its trust store.
3. The Policy Management server creates a local certificate and exports the certificate to the CMP system.
4. The CMP system imports the peer certificate (local certificate created by the Policy Management server) into its trust store.

Note: Procedures used in this chapter may require the reboot of one or more servers. Subsequently, for high availability (HA) to operate correctly in a clustered system, the active server of the cluster must not be rebooted unless the cluster is in the **online** state. Before rebooting any server, check cluster status using the CMP interface. If a cluster is labeled **Degraded**, but the server detail does not show any failed or disconnected equipment, the server is performing a database synchronization operation and until the synchronization process has completed, the standby server cannot perform as the active server.

When a new certificate is configured, the synchronization causes the HA on the standby server to restart.

SSL certificates are created on a per-cluster basis, and to ensure that the cluster has the same certificate installed, you should force a system synchronization.

To exchange certificates in a large Policy Management network with many servers, see Bulk Certificate Exchange.

Exporting a Local Certificate to a Policy Management Server

To export a local certificate through a secure connection between the CMP system and a Policy Management server:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.

- To access the platcfg utility from the system console:
  a. Log in as **root**.
  b. Enter `su - platcfg`.
- To access the platcfg utility through an SSH remote session:
  a. Log in as **admusr**.
  b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **SSL Key Configuration** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Configure Keystore** from the Configure SSL keys Menu screen and press **Enter**.
5. Select **Export key** from the Operate keystore Menu screen and press **Enter**.
6. Enter the **Keystore Password**, select **OK**, and press **Enter**.
7. Press **Enter** to accept the alias `tomcat`.

   The Export Certificate screen opens.
8. Select the certificate type **binary**, enter the local certificate file path, select **OK**, and press **Enter**.

   The certificate is exported.
9. When the certificate is exported, a successful completion message displays.

   Press **Enter**.
10. Log in as **admusr** on the active server of the CMP cluster and enter the following commands:
    a. `sudo su -`
    b. `scp admusr@`*`active_server_addr`*`:`*`remote_path/file`*`.cer` *`local_path`*

    In this example, *active_server_addr* is `mpe-01`, *remote_path* is `/tmp`, *file* is `mpe-a.cer`, and *local_path* is `/tmp`:

```
# scp admusr@mpe01:/tmp/mpe-a.cer /tmp
mpe-a.cer
#
```

    The certificate is copied to the active CMP server.

Importing a Peer Certificate

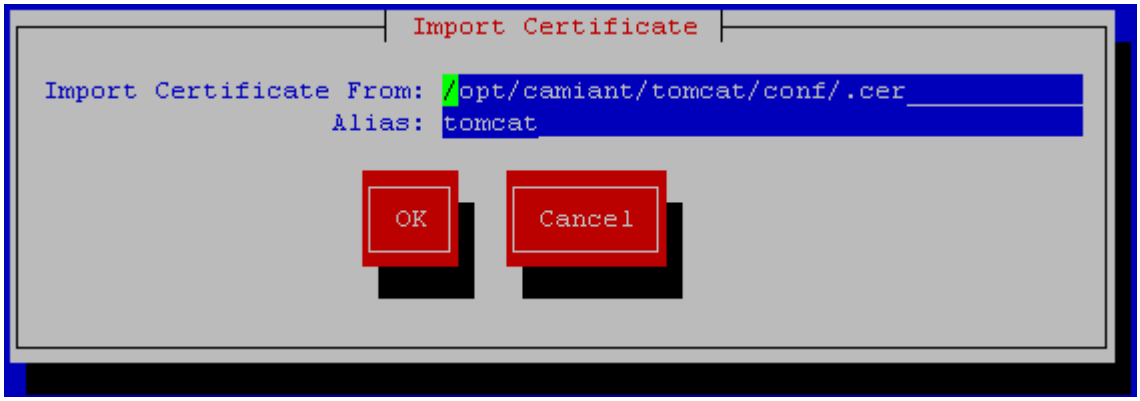
This procedure imports a certificate to a Policy Management server and enables a secure connection. This includes certificates generated by other servers including certificates signed by a third party or similar.

After you have exported the local certificate, to import the peer certificate (that is, the certificate you exported) to the certificate keystore of a Policy Management server:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **SSL Key Configuration** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Configure cacerts** from the Configure SSL keys Menu screen and press **Enter**.
5. Select **Import trusted key** from the Operate keystore Menu screen and press **Enter**.
6. Enter the **Keystore Password**, select **OK**, and press **Enter**.
7. Enter the import location and **Alias** for the certificate, as set previously for the **CN** name, select **OK**, and press **Enter**.

Figure 2-10 Import Certificate



You are then presented with the certificate data for verification. Ensure that the **CN** name, **Owner**, and **Issuer** names of the input file name match that of the previous export file.

8. After you have verified that the certificate data is correct, select **OK** and press **Enter**.

   When the certificate is imported, a successful import message displays.

9. Press **Enter**.

10. Select **Cluster File Sync** from the Policy Configuration Menu screen and press **Enter**.

    The imported peer certificate is synchronized to the others servers of the cluster.

11. Select **Restart Application** from the Policy Configuration Menu screen and press **Enter**.

    The Policy Management application (the `qp_procmgr` process) on the active server restarts.

    Tip: You can verify that SSH keys have been fully exchanged between servers by logging in to the active CMP server as **admusr** and entering the following commands:

    ```
    sudo su - /opt/camiant/bin/qpSSHKeyProv.pl --check --verbose
    ```

Once certificates are exchanged, to enable an HTTPS connection to the Policy Management cluster, log on to the active CMP server, select the cluster, select the **Secure Connection** check box from the **Policy Server** tab, and click **Save**. You are prompted, "`The configuration was applied successfully`," and **Secure Connection** displays **Yes**. See the appropriate *CMP User's Guide* for more information.

Tip: If instead you are prompted that the Policy server is unavailable, there may be a problem with the certificates.

Bulk Certificate Exchange

Before beginning this procedure, you must have created self-signed certificates (see Creating a Self-signed Certificate).

This procedure imports certificates from multiple MPE andMRA clusters and enables a secure connection. You would use this procedure, in place of the procedures Exporting a Local Certificate to a Policy Management Server and Importing a Peer Certificate, to save time when exchanging certificates in a large Policy Management network.

You cannot use this procedure for connections between a Network Configuration Management Platform (NW-CMP) system and a System Configuration Management Platform (S-CMP) system.

From the primary site active CMP or S-CMP server:

1. Log in as **admusr**.

2. Enter `sudo su -`.

3. To exchange SSH keys between the CMP system and MPE and MRA servers, enter `/opt/camiant/bin/qpSSHKeyProv.pl --prov --relax`.

   The argument `--relax` causes SSH keys to be provisioned from MPE and MRA systems to the CMP system.

4. Enter `/opt/camiant/bin/qpRunInTopo.py --cmd="sslKeyUtil --exportToCmp --target=`*`active_cmp_addr`*`" --pool-size=1 --prod=mpe,mra --ha-role=Active [--show]`.

   The optional argument `--show` displays execution details.

   The utility `sslKeyUtil` executes on the active server of each MPE and MRA cluster. It exports the certificate from the local keystore to a local file; copies the file to the specified CMP server; and imports the file into the certificate keystore on the CMP server.

5. Synchronize the certificates across the other servers in the CMP cluster. For more information, see Synchronizing Cluster Files.

Example 2-1 Example

This example shows a successful execution of `qpRunInTopo.py`. The certificate file `mpe-a.cer` is imported from the MPE server `mpe01` to the active CMP server at IP address `nn.nn.nn.nn`.

```
# /opt/camiant/bin/qpRunInTopo.py --cmd="sslKeyUtil --exportToCmp --
target=nn.nn.nn.nn" --pool-size=1 --prod=mpe,mra --ha-role=Active --show
Command will be run on following servers:
["mpe01"]
Continue? [yes|no]: yes
[   {   'errput': 'FIPS integrity verification test failed.\r\nCertificate
stored in file </tmp/mpe01_mpe-a.cer>\n',
        'id': 'admusr@mpe01: sslKeyUtil --exportToCmp --target=nn.nn.nn.nn',
        'output': 'Export to cmp\n\Going to export key mpe-a\n\Importing to
cacerts.jks in target nn.nn.nn.nn\nSSHRun returns 0\n',
        'ret_code': 0}]
=======================================
Suceeded.
#
```

Once certificates are exchanged, to enable an HTTPS connection, log on to the active CMP server, select the Policy Management cluster, and select the **Secure Connections** check box, located on the **Policy Server** tab. See the appropriate *CMP User's Guide* for more information.

About Creating CA Third-Party Signed Certificates

Note: This section assumes that no SSL certificates have previously been generated on or imported into the servers. If pre-existing certificates exist on the system (besides the default `tomcat` certificate, which you must keep), contact My Oracle Support to determine their purpose and importance. Read this section in its entirety before starting the operations.

To create CA third-party certificates, execute the following procedures:

1. Deleting an SSL Certificate

2. Generating a Certificate Signature Request
3. Exporting the Certificate Signature Request from the System
4. Provide the Certificate Signature Request to the third party who signs and returns the certificate request.
5. Importing Third-party Peer Certificates
6. Importing the Third-party Signed Certificates
7. Synchronizing and Rebooting the Cluster

Deleting an SSL Certificate

Note: You can also use this procedure to delete an expired SSL certificate.

Before continuing with any of the other required certificate generation or import/export functions, delete any other user-created pre-existing certificates.

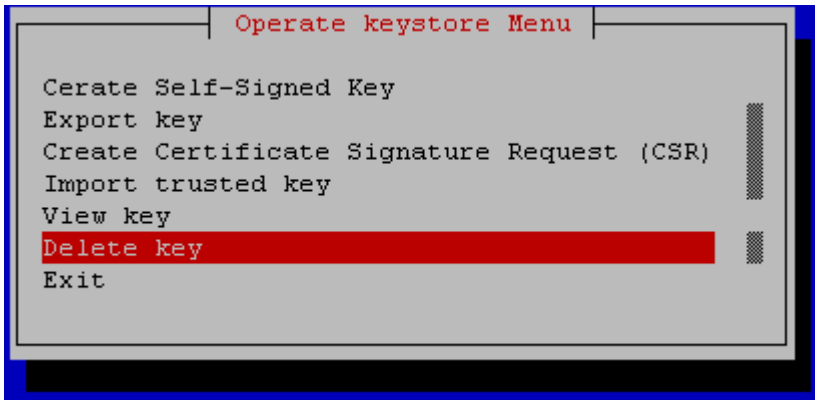
Caution: The default certificate has the alias `tomcat`. You may need to replace it with a current certificate, but do not delete it, or else you will not be able to complete subsequent procedures.

To delete an SSL certificate:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

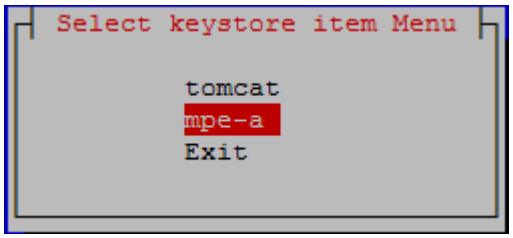
   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **SSL Key Configuration** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Configure Keystore** from the Configure SSL keys Menu screen and press **Enter**.
5. Select **Delete key** from the Operate keystore Menu screen and press **Enter**.

Figure 2-11 Operate keystore Menu



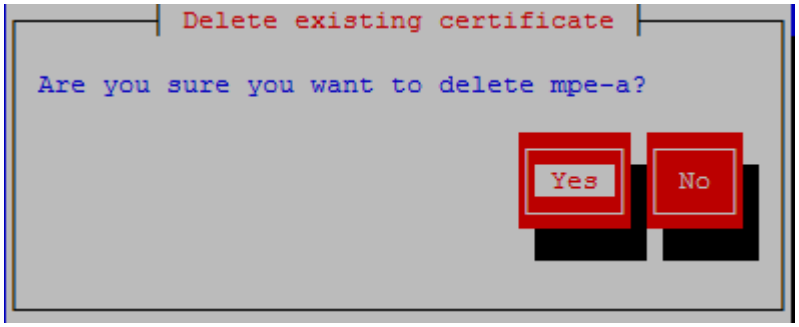
6. Enter the **Keystore Password**, select **OK**, and press **Enter.**
7. Select the certificate to be deleted and press **Enter**.

Figure 2-12 Select keystore item Menu



8. You are prompted to delete the selected certificate.

Figure 2-13 Delete existing certificate



Select **Yes** to delete the certificate or **No** to leave it as is, then press **Enter**.

You are now ready to generate the local certificate, export it for signing, and then re-import the signed certificate.

About Generating a Certificate Signature Request

To generate the third-party signed local certificate, execute the following procedures:

1. Generating a Certificate Signature Request
2. Exporting the Certificate Signature Request from the System
3. Send the Certificate Signature Request to a third-party certifying authority for signing

4. Receive the signed Certificate Signature Request
5. Importing the Third-party Signed Certificates
6. Verifying a Self-signed Certificate

Generating a Certificate Signature Request

To generate a certificate signature request:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu and press **Enter**.
3. Select **SSL Key Configuration** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Configure keystore** from the Configure SSL keys Menu screen and press **Enter**.
5. Select **Create Certificate Signature Request (CSR)** from the Operate keystore Menu screen and press **Enter**.
6. Enter the **Keystore Password**, select **OK**, and press **Enter**.
7. Select the certificate name that you want to export for signature from the Create CSR screen and press **Enter**.

   Note: The alias of the certificate is used later for re-importing the certificate after signing by a third party. Use an alias that allows the certificate to be identified with a specific system. Also of importance is the **Expiration** attribute, which should be set to a sufficiently large value so that the certificate does not expire before any peer certificates. Oracle recommends a value preventing expiration for three years.
8. Edit the destination path from the Create CSR screen to change it or select **OK** to accept it, and press **Enter**.

Exporting the Certificate Signature Request from the System

To export a locally generated certificate signature request:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **SSL Key Configuration** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Configure keystore** from the Configure SSL keys Menu screen and press **Enter**.
5. Select **SSL Key Configuration** from the Policy Configuration Menu screen and press **Enter**.
6. Select **Export key** from the Operate keystore Menu screen and press **Enter**.
7. Enter the **Keystore Password**, select **OK** and press **Enter**.
8. Select the certificate to export for signature and press **Enter**.
9. Select the certificate **ascii** and enter the certificate export location, then select **OK** and press **Enter**.

   The **Message** screen opens to confirm that the certificate was exported.

After the certificate file is exported, send it to the third party who signs and returns the certificate request.

Importing the Third-party Signed Certificates

After the certificate has been signed by the third-party certifying authority, two certificate files are returned by the authority for importing into the Policy Management servers:

- A signed local client certificate (with the file suffix `.crt`)
- A certificate authority (**CA**) peer certificate (with the file suffix `.pem`)

Both certificates must be imported into the active CMP system for proper SSL communication.

Note: It may necessary to edit the returned files to remove extraneous debugging information in the certificate. You must use a Linux-based editor to preserve line termination style.

The only content in the files should be the blocks of data beginning with:

```
----BEGIN CERTIFICATE-----
```

and ending with:

```
-----END CERTIFICATE-----
```

All other text above or below these blocks should be removed.

A further modification needs to be made to the signed local client certificate.

For the Policy Management servers to be able to import the local certificate successfully, the CA peer certificate must be merged into the signed local client certificate. Copy the `BEGIN/END` certificate text block from the CA peer certificate into the local client certificate below the `BEGIN/END` certificate text block. The final result is the original local client certificate text block immediately followed by the certificate text block of the CA peer certificate that was provided by the third-party signer. An example of what this should look like is as follows:

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAligAwIBAgIBBTANBgkqhkiG9w0BAQUFADCBjDELMAkGA1UEBhMCVVMx
<text removed>
gJeTRnZwMJEXv71V85NGobVGqb1uR94kIQazFP5HC2b2C0Q=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIDjTCCAvagAwIBAgIJAJCKgXrXbhQ/MA0GCSqGSIb3DQEBBQUAMIGMMQswCQYD
<text removed>
YVPOATiFnrt1B9Qb1P8kW8lwPmG88Gg6nqttolhAnIi/lWBcp+QZfJMxPBcMkH2k7A==
-----END CERTIFICATE-----
```

Either copy these certificate files to the Policy Management server in advance, or store them somewhere on the network accessible via SCP. They can be imported back into the system to secure the communication channel with the third-party system.

To import the certificates:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **SSL Key Configuration** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Configure keystore** from the Configure SSL keys Menu screen and press **Enter**.
5. To import the local signed certificate, select **Import trusted key** from the Operate keystore Menu screen and press **Enter**.
6. Enter the **Keystore Password**, select **OK**, and press **Enter**.

   You are prompted for the location of the certificate to be imported.
7. Select or enter the location where the local signed certificate is located and the certificate alias name, select **OK**, and press **Enter**.

   The certificate data screen opens for verification. To avoid confusion, though they may be different, ensure that the **Owner** and **Issuer** names used for the certificate match the host name of the server where the certificate is being created.

   Note: The alias entered must match the alias originally used to create the certificate request.
8. To import the CA signed certificate as a peer certificate, select **Import trusted key** from the Operate cacerts Menu and press **Enter**.
9. Enter the **Keystore Password**, select **OK**, and press **Enter**.

   You are prompted for the location of the certificate to be imported.
10. Select or enter the location where the CA peer certificate is located and the certificate alias name, select **OK**, and press **Enter**.

    The certificate data screen opens for verification. To avoid confusion, though they may be different, ensure that the **Owner** and **Issuer** names used for the certificate match the host name of the server the certificate is being created on. If all certificate information is correct, the next operation is to import the CA certificate as a peer certificate.

    Note: The alias entered must match the alias originally used to create the Certificate Request.

Importing Third-party Peer Certificates

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:

a. Log in as **admusr**.

b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

---

2. Select **Policy Configuration** from the Main Menu and press **Enter**.
3. Select **SSL Key Configuration** from the Policy Configuration Menu and press **Enter**.
4. Select **Configure keystore** from the Configure SSL keys Menu and press **Enter**.
5. Select **Import trusted key** from the Operate keystore Menu and press **Enter**.
6. Enter the **Keystore Password**, select **OK**, and press **Enter**.

   You are prompted for the location of the certificate to be imported.

7. Select or enter the location where the certificate is located and the certificate alias name, select **OK**, and press **Enter**.

Note: The alias entered here must match the alias originally used to create the Certificate Request.

Synchronizing and Rebooting the Cluster

After exchanging certificates, all cluster servers must be synchronized and rebooted.

- Synchronizing a cluster shares the keystore. To synchronize, see Synchronizing Cluster Files.
- To reboot, see the *CMP User's Guide* corresponding to the system mode.

3

Synchronizing Files

This chapter describes how and when to synchronize files in clusters.

Files should be synchronized after either of the following items are configured:

- Routes (Routing Config)
- Firewall (Firewall)

Managing Cluster Sync Configurations

Use the **Cluster Sync Config** menu to manage cluster sync configurations:

- Reading Destination from COMCOL
- Adding Sync File
- Deleting Sync File

Reading Destination from COMCOL

To read the cluster sync destination from COMCOL:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Cluster File Sync** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Cluster Sync Config** from the Cluster Configuration Sync Menu screen and press **Enter**.

5. Select **Read Destination from Comcol** from the Config the Destination of Cluster Sync Menu screen and press **Enter**.

   The destination of the cluster sync file is read from COMCOL.

Adding a Sync File

To add a cluster sync configuration file:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.

   - To access the platcfg utility from the system console:

     a. Log in as **root**.

     b. Enter `su - platcfg`.

   - To access the platcfg utility through an SSH remote session:

     a. Log in as **admusr**.

     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.

3. Select **Cluster File Sync** from the Policy Configuration Menu screen and press **Enter**.

4. Select **Cluster Sync Config** from the Cluster Configuration Sync Menu screen and press **Enter**.

5. Select **Add Sync File** from the Config the Destination of Cluster Sync Menu screen and press **Enter**.

   The Add a Sync File screen opens.

6. Enter data into the fields:

   - **filename**—The name of the sync file.
   - **remote file**—The name of the sync file if different at the remote site.
   - **scope (cluster/site/clusterGroup)**—Lists where each file is to be synced:
     - **cluster**—Indicates access to all servers at all sites. Files that need to be in sync at all sites (such as certificates) should be listed as Cluster.
     - **site**—Indicates access to servers at the local site. IP-related files that may not be valid at other sites (such as firewall and static routes) should be listed as Site.
     - **clusterGroup**—Indicates access to all servers only in multiple CMP, MPE, or MRA clusters.

- **post script**— Indicates a note or description of the scope (cluster/site/clusterGroup).

7. Select **OK** and press **Enter**.

   The new cluster sync configuration is saved.

Deleting a Sync File

To delete an cluster sync configuration file:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Cluster File Sync** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Cluster Sync Config** from the Cluster Configuration Sync Menu screen and press **Enter**.
5. Select **Delete Sync File** from the Config the Destination of Cluster Sync Menu screen and press **Enter**.

   The Main Routing Table screen opens.

6. Select the cluster sync configuration file to delete from the list, select **OK** and press **Enter**.

   The selected cluster sync configuration is deleted.

Displaying a Sync Configuration

Displaying a sync configuration is useful when georedundancy is implemented.

To display a sync configuration:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:

a. Log in as **root**.

b. Enter `su - platcfg`.

- To access the platcfg utility through an SSH remote session:

a. Log in as **admusr**.

b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

---

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Cluster File Sync** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Show Sync Config** from the Cluster Configuration Sync Menu screen and press **Enter**.

   The **Sync File** screen is displayed.

   The **Scope** column lists where each file is being synced:

   - **Site** indicates that the file is synced to servers at the local site
   - **Cluster** indicates that the file is synced to all servers at all sites

---

Note: Files that must be in sync at all sites (like certificates) are listed as **Cluster**; IP-related files that are not valid at other sites (like firewall and static routes) are listed as **Site**.

Displaying a Sync Destination

To display a sync destination (for example, hostname, IP address, and location):

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.

   - To access the platcfg utility from the system console:

   a. Log in as **root**.

   b. Enter `su - platcfg`.

   - To access the platcfg utility through an SSH remote session:

   a. Log in as **admusr**.

   b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

---

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.

3. Select **Cluster File Sync** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Show Sync Destination** from the Cluster Configuration Sync Menu screen and press **Enter**.

   The Sync Destination screen opens.

Displaying a Sync Status

To display a cluster sync status:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Cluster File Sync** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Show Sync Status** from the Cluster Configuration Sync Menu screen and press **Enter**.

   A screen opens to display the sync status.

Synchronizing Cluster Files

Note: File synchronization (or cluster sync) copies configuration files from the target server to the remaining servers in the cluster. Performing a cluster sync will launch **`qp_procmgr`** on the target servers, so this action should only be performed from the Active server, or else a failure occurs. A warning screen opens before continuing with the sync to help prevent this issue from occurring. There is a separate sync operation for DSCP configurations.

To synchronize the cluster files:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:

a. Log in as **root**.

b. Enter `su - platcfg`.

- To access the platcfg utility through an SSH remote session:

  a. Log in as **admusr**.

  b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

---

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Cluster File Sync** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Start Synchronizing** from the Cluster Configuration Sync Menu screen and press **Enter**.

Note: A warning message screen opens, indicating that a cluster sync will launch `qp_procmgr` on the target servers.

WARNING: This action should only be performed from the Active server, otherwise a failure will occur.

---

5. Select **OK** and press **Enter**.

   Configuration files are synced to the other servers in the cluster, and `qp_procmgr` is restarted on the target servers.

4

Backing Up and Restoring the System and Server

This chapter describes how to back up and restore the system and server.

Backing Up a Server

The server backup file contains OS-level information that is configured in the platcfg utility such as **IP**, **NTP**, and **DNS** addresses. This type of backup is unique to a server and should be created for every server within a cluster.

To back up a server:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select the **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.

   System Backup and **System Restore** actions are only allowed on a CMP server or **MA Server**, so these options are not available on the menu for other types of servers.
4. Select **Server Backup** from the Backup and Restore Menu screen and press **Enter**.
5. Accept the default backup directory or enter the **ISO** path to save the backup file.

   The naming convention used for the backup file is:

   hostname-camiant-*release*-serverbackup-*datetime*.iso

6. Select **OK** and press **Enter**.

   The backup file is created.

Backing Up the System

The system backup file contains application-level information such as Topology, Network Element, and Policy Management configurations that are configured in the CMP system. This backup file saves the information for an entire deployment and should be created on the active server of the Primary CMP cluster.

When the backup file is created, the file contains a specific name and is located in a specific directory. Transfer this backup to the **FTP** server and the **PMAC** server.

To back up the system:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
4. Select **System Backup** from the Backup and Restore Menu screen and press **Enter**.
5. Enter the tar.gz path to save the backup file.
6. Accept the default backup directory or enter a desired directory.

   The naming convention used for the backup file is:

   hostname-camiant-*release*-systembackup-*datetime*.tar.gz
7. Select **OK** and press **Enter**.

   The backup file is created.

Displaying Backup Files

To display current local archive and remote archive backup files:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.

- To access the platcfg utility from the system console:
  a. Log in as **root**.
  b. Enter `su - platcfg`.
- To access the platcfg utility through an SSH remote session:
  a. Log in as **admusr**.
  b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select the **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Display Backup Files** from the Backup and Restore Menu screen and press **Enter**.
5. From the Display Backup Files Menu screen, select either the local archive or the remote archive:
   - Select **Display Local Archive** and press **Enter**.
     The Local Archives screen opens.
   - Select **Display Remote Archive** and press **Enter**.
     The Remote Archives screen opens.

Configuring Local Archive Settings

You can store up to three archives for both the server and system backup files. To configure local archive settings:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.

3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Local Archive Settings** from the Backup and Restore Menu screen and press **Enter**.
5. Specify the number of archives for the server and system backups.

   Note: The server backup option is only available on a CMP system or MA Server.

6. When finished, select **OK** and press **Enter**.

   The archive settings are configured.

Configuring Remote Archive Settings

This section describes how to manage remotely stored system and server archives.

Adding a Remote Archive

To add a remote archive:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Remote Archive Settings** from the Backup and Restore Menu screen and press **Enter**.
5. Select **Remote Archive for Server Backups** or **Remote Archive for System Backups** from the Remote Archive Settings Menu screen and press **Enter**.

   Note: The server backups option is available only on a CMP system or MA Server.

6. Select **Add Remote Archive** from the second Remote Archive Settings Menu screen and press **Enter**.
7. Enter the remote access information, where:
   - **user** and **password**—Valid SSH login credentials for the target server.
   - **host**—A reachable IP address or a resolvable hostname.
   - **folder**—A directory on the target server where the Policy Management server will attempt to copy backups. The directory must already exist; it will not be created on demand.
   - **comment**—The name of the remote archive when viewed in platcfg.
8. Select **OK** and press **Enter**.

   The remote archive is added.

Editing a Remote Archive Configuration

To edit a remote archive configuration:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select either **Remote Archive for Server Backups** or **Remote Archive for System Backups** from the Remote Archive Settings Menu screen and press **Enter**.
3. Select **Edit Remote Archive** from the Remote Archive Settings Menu screen and press **Enter**.
4. Select the remote archive to edit from the Remote Archives Menu screen and press **Enter**.
5. Enter the remote archive information, select **OK**, and press **Enter**.

   The remote archive settings are updated.

Deleting a Remote Archive Configuration

To delete a remote archive configuration:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select either **Remote Archive for Server Backups** or **Remote Archive for System Backups** from the Remote Archive Settings Menu screen and press **Enter**.
3. Select **Delete Remote Archive** and press **Enter**.
4. Select the remote archive to delete from the Remote Archives Menu screen and press **Enter**.
5. Select **Yes** from the Confirm deletion screen and press **Enter**.

   The remote archive is deleted.

Displaying a Remote Archive Configuration

To display a remote archive configuration:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select either **Remote Archive for Server Backups** or **Remote Archive for System Backups** from the Remote Archive Settings Menu screen and press **Enter**.
3. Select **Display Remote Archive** from the second Remote Archive Settings Menu screen and press **Enter**.

Either the **Display Remote Archive For Server-Backup** or the **Display Remote Archive For System-Backup** screen opens.

Scheduling Backups

You can configure your system or server to conduct backups on a scheduled basis. This section describes how to manage backup schedules.

Scheduling a Backup

To schedule a backup:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Scheduled Backup Settings** from the Backup and Restore Menu screen and press **Enter**.
5. Select either **Scheduled Backup for Server Backups** or **Scheduled Backup for System Backups** from the Scheduled Backup Settings Menu screen and press **Enter**.
6. Select **Add Schedule** from the **Scheduled Backup for server backups Menu** screen and press **Enter**.

   The Schedule parameters screen opens.
7. Enter the following information:
   - **Name**—A unique name identifying the scheduled backup.
   - **Min**—Minute to perform backup. Valid values are 0 to 59, with a default of 0.
   - **Hour**—Hour to perform backup. Valid values are 0 to 23, with a default of 0.
   - **Weekly**—Select to have the backup performed weekly. When **Weekly** is selected, the **Days of the Month** value is ignored. The default backup is performed weekly.

- **Days of Week**—Specifies that the backup is performed on specific days. Valid values are sun, mon, tue, wed, thu, fri, and sat.
- **Monthly**—Select to have the backup performed monthly. When **Monthly** is selected, the **Days of the Week** value is ignored.
- **Days of the Month**—Day to perform backup. Valid values include 1 and 15.

Note: When **Weekly** is selected, the **Days of the Month** field is ignored, and when **Monthly** is selected, the **Days of the Week** field is ignored.

8. Select **OK** and press **Enter**.

   The backup is scheduled.

Editing a Backup Schedule

To edit a backup schedule:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select the **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Scheduled Backup Settings** from the Backup and Restore Menu screen and press **Enter**.
5. Select either **Scheduled Backup for Server Backups** or **Scheduled Backup for System Backups** from the Scheduled Backup Settings Menu screen and press **Enter**.
6. Select **Edit Schedule** from the **Scheduled Backup for server backups Menu** screen or from the Scheduled Backup for system backups Menu screen and press **Enter**.
7. Edit the following Information:
   - **Name**—A unique name identifying the scheduled backup.
   - **Min**—Minute to perform backup. Valid values are 0 to 59, with a default of 0.

- **Hour**—Hour to perform backup. Valid values are 0 to 23, with a default of 0.
- **Weekly**—Select to have the backup performed weekly. When **Weekly** is selected, the **Days of the Month** value is ignored. The default backup is performed weekly.
- **Days of Week**—Specifies that the backup is performed on specific days. Valid values are sun, mon, tue, wed, thu, fri, and sat.
- **Monthly**—Select to have the backup performed monthly. When **Monthly** is selected, the **Days of the Week** value is ignored.
- **Days of the Month**—Day to perform backup. Valid values include 1 and 15.

Note: When **Weekly** is selected, the **Days of the Month** field is ignored, and when **Monthly** is selected, the **Days of the Week** field is ignored.

8. Select **OK** and press **Enter**.

Deleting a Backup Schedule

To delete a backup schedule:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
5. Select **Scheduled Backup Settings** from the Backup and Restore Menu screen and press **Enter**.
6. Select either **Scheduled Backup for Server Backups** or **Scheduled Backup for System Backups** from the Scheduled Backup Settings Menu screen and press **Enter**.

7. Select **Delete Schedule** from the **Scheduled Backup for server backups Menu** screen or from the Scheduled Backup for system backups Menu screen and press **Enter**.
8. Select **OK** and press **Enter**.

   The schedule is deleted.

Displaying a Backup Schedule

To a display a backup schedule:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.
   - To access the platcfg utility from the system console:
     a. Log in as **root**.
     b. Enter `su - platcfg`.
   - To access the platcfg utility through an SSH remote session:
     a. Log in as **admusr**.
     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.
2. Select **Policy Configuration** from the Main Menu and press **Enter**.
3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
5. Select **Scheduled Backup Settings** from the Backup and Restore Menu screen and press **Enter**.
6. Select **Display Scheduled Backups** from the Scheduled Backup Settings Menu screen and press **Enter**.

   The backup schedule list screen opens.

Restoring a System

Restoring a System restores the Policy Management information that is unique to this system, including topology, policies, and feature configuration.

To restore a system:

1. Stop QP and COMCOL on the standby server using the CMP interface, by entering the commands:

```
service qp_procmgr stop
service comcol stop
```

2. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.

   - To access the platcfg utility from the system console:

     a. Log in as **root**.

     b. Enter `su - platcfg`.

   - To access the platcfg utility through an SSH remote session:

     a. Log in as **admusr**.

     b. Enter `sudo su - platcfg`.

   Note: The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

3. Select the **Policy Configuration** from the Main Menu screen and press **Enter**.

4. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.

5. Select **System Restore** from the Backup and Restore Menu screen and press **Enter**.

6. Input the requested information, where:.

7. Select **OK** and press **Enter**.

   The system restores to the backup version specified.

8. Restart QP and COMCOL on the standby server using the CMP interface, by entering the commands:

```
service comcol start
service qp_procmgr start
```

Note: For more information about how to use the CMP interface, refer to the *CMP User's Guide* that corresponds to the mode of the system.

Performing a Server Restore

The server restore restores the OS information unique to the server. This operation applies the data from a previously saved server configuration backup file.

To perform a server restore:

1. Log in to the platcfg utility using one of two methods, either from the system console using **root** or through an SSH remote session using **admusr**.

   - To access the platcfg utility from the system console:

a. Log in as **root**.

b. Enter `su - platcfg`.

- To access the platcfg utility through an SSH remote session:

  a. Log in as **admusr**.

  b. Enter `sudo su - platcfg`.

> **Note:** The dash (-) is required in the `su - platcfg` or the `sudo su - platcfg` command to ensure proper permissions.

2. Select **Policy Configuration** from the Main Menu screen and press **Enter**.
3. Select **Backup and Restore** from the Policy Configuration Menu screen and press **Enter**.
4. Select **Server Restore** from the Backup and Restore Menu screen and press **Enter**.
5. Enter the path to the backup file, select **OK**, and press **Enter**.

   The system restores to the backup version specified.