

Core and Android and iOS Client Mobile Device Management Protection Profile Guide for Release 11

August 2021

Contents

What is Common Criteria mode?	4
Target of Evaluation	4
Accessing administrative interfaces	4
Device types in Common Criteria evaluation	4
Connecting to Core from the CLI	6
Logging in	6
Logging out	6
Help commands	6
How to deploy Core in Common Criteria mode	7
After installation or upgrade tasks	8
Configuring your external services	8
Disabling SSH and CLI	9
Configuring Transport Layer Security (TLS)	9
Trusting certificates for connections to your external services	9
Importing x509v3 certificates as trusted root certificates	10
Enabling FIPS mode	11
Enabling Common Criteria mode	12
Enabling Common Criteria mode for Samsung Knox devices	12
Uploading a valid, trusted, FIPS-compatible certificate	14
Selecting cipher suites	15
Changing the port settings	16
Configuring LDAP servers	17
Configuring mutual authentication	21
Setting the EULA or other login text	26
Setting up Core with a closed network / AOSP deployment	27
Use cases for AOSP deployments	27
Enabling a closed network / AOSP deployment in Core	28
Creating a new Android Enterprise configuration	30
Provisioning the Android device	31
Managing the closed network / AOSP devices	32
General Core configuration tasks	35
Determining your Core version	35
Configuring administrators for roles defined by federal requirements	36
Configuring the Admin Portal timeout	41
Configuring certificate authentication for password-less access	42
Setting up PIN registration	42
Configuring the sync interval for a device	42
Limiting the number of devices that users can enroll	43
Checking registration and check-in/connectivity status	43
Querying hardware and software information about a device	44
Querying the installed apps on a device	44
Installing policies on a device	44

Using labels for application groups	45
Configuring an ECDSA client identity certificate for mutual authentication	51
Updating Core	53
Provisioning AE devices in a closed network or AOSP deployment	56
Understanding AE device provisioning requirements	56
Creating a new Android Enterprise configuration	58
Disabling the QR code and registration URL	59
Configuring Knox mobile enrollment	60
Prerequisites for Knox mobile enrollment	60
Creating the Knox mobile enrollment profile	61
Assigning the KME profile to devices	64
Android-related configuration tasks	66
Deploying Mobile@Work for Android	66
Configuring the Android warning banner	75
Configuring the lockdown policy for Android devices	76
Configuring allowed app sources for Android Samsung Knox devices	76
Wi-Fi settings for Android devices	79
Disabling the Developer options menu on Android devices	85
Prohibiting device users from unenrolling	86
Disabling biometric authentication on Android devices	86
Quarantining an Android device based on its OS version	87
iOS-related configuration tasks	90
Configuring the password policy for iOS devices	90
Configuring iOS restrictions	90
Wi-Fi settings for iOS devices	91
Configuring VPN networks on iOS devices	94
Updating the operating system for iOS devices	94
Setting up the Apps@Work web clip	95
Managing devices	96
Device tasks in other sections	96
Registering/Enrolling a device	97
Locking a device	98
Wiping a device	98
Unenrolling a device	98
Sending a message to a device	98
The App Catalog	99
Installing and removing apps on a device	100
Collecting, viewing, and exporting logs	101
Viewing audit log information	101
Exporting Admin Portal audit logs	102
Exporting device status events	123
Collecting audit events for Android devices	126
Exporting System Manager audit logs	135

What is Common Criteria mode?

Common Criteria mode refers to a set of features in Core that meet requirements associated with Common Criteria. Also referred to as Common Criteria for Information Technology Security Evaluation, Common Criteria is an international set of guidelines and specifications for evaluating information security products to ensure they meet the established security standard for government deployments.

For a detailed description of the Common Criteria evaluation, see the *MobileIron Platform 11 Security Target*.

Target of Evaluation

The target of evaluation is:

- Core 11 server, deployed as a virtual machine (VM).
 - The virtual deployments are in VMWare ESXi (6.7U2, 7.0)
- Mobile@Work 11 for Android

Accessing administrative interfaces

During setup, two local users having the same credentials are created, one for the Admin Portal and one for the System Manager. If you make changes to the roles or password for the Admin Portal user, the changes do not affect the System Manager user.

Use the following URLs to access the configuration screens in this document.

- **Admin Portal:** `https://<fully-Qualified-Domain-Name>/admin`
- **System Manager:** `https://<fully-Qualified-Domain-Name>:8443/admin`
- **User Portal:** `https://<fully-Qualified-Domain-Name>/user`



The CLI is available only through a VMWare console.

Device types in Common Criteria evaluation

- ["Android device types" on the next page](#)
- ["iOS device types" on the next page](#)
- ["Other device platforms" on the next page](#)

Android device types

The Common Criteria evaluation for Core 11 includes the following Android devices:

Android 10

- Samsung Galaxy S20 5G - SM-G981U1 -
- Samsung Galaxy Note 10 - SM-N970U1
- Samsung Galaxy A71 5G - SM-A716V

Android 11

- Samsung Galaxy S21 Ultra 5G - SM-G998U

iOS device types

The Common Criteria evaluation for Core 11 includes no iOS device agent security function claims. The iOS devices themselves have been evaluated separately under NIAP VID 11036. However, the Common Criteria evaluation for Core supports the enrollment and subsequent management of the following:

- iOS devices running iOS 13.5:
 - iPad (7th Gen)
 - iPhone 11

Other device platforms

Interaction with Windows 10 and macOS devices is unchanged when Core is in Common Criteria mode. However, Windows 10 and macOS devices are not included in the Common Criteria evaluation.

Connecting to Core from the CLI

The CLI, or command line interface, enables authorized administrators to access certain functions from the command line in a terminal window.

Logging in

To log into Core securely, use the network protocol SSH, also known as Secure Shell or Secure Socket Shell. SSH gives users, particularly system administrators, a secure way to access a computer over an unsecured network.

Procedure

1. Use SSH to make the connection to Core.
2. Log in as the Administrator user established during installation.
3. Enter the password and click **Enter**.

Logging out

Enter **Ctrl-d** to terminate the CLI session and close the terminal window. You can also enter one of the following commands:

- **logout**
- **exit**

Help commands

Two commands are available to help you use the CLI:

- **help**
- **?** (question mark)

Enter **help** to display a description of the interactive help system, including:

- Auto-complete keys
- Movement keys
- Deletion keys

Enter **?** to list available commands in the current mode or details for the current command. For example, the following command lists all commands in the current mode:

>?

How to deploy Core in Common Criteria mode

After installation or upgrade tasks	8
Configuring your external services	8
Disabling SSH and CLI	9
Configuring Transport Layer Security (TLS)	9
Trusting certificates for connections to your external services	9
Importing x509v3 certificates as trusted root certificates	10
Enabling FIPS mode	11
Enabling Common Criteria mode	12
Enabling Common Criteria mode for Samsung Knox devices	12
Uploading a valid, trusted, FIPS-compatible certificate	14
Selecting cipher suites	15
Changing the port settings	16
Configuring LDAP servers	17
Configuring mutual authentication	21
Setting the EULA or other login text	26

For first-time installations, complete the following tasks to install Core.



Core release 11 is used as an example in this section, but you may need to replace the example URLs with your specific release number.

Procedure

1. Complete the steps in the “Pre-Deployment Tasks” chapter in the *On-Premise Installation Guide for Core and Enterprise Connector*.



Port 7443 is open on Core by default. It is available for use with the Reporting Database. It is also the recommended port for the Apps@Work port when mutual authentication is enabled and you are using the Apps@Work web clip with certificate authentication on iOS devices.

2. Complete the steps in the “Core Installation” chapter in the *On-Premise Installation Guide for Core and Enterprise Connector*.

3. See your customer service representative for the correct Core ISO file or Core repository ZIP file for your release.



If you are using a repository, first create a directory for the new Core version and unzip the Core repository ZIP file into that directory. Do not overwrite the files from the previous Core version.

After installation or upgrade tasks

Complete the following tasks after installation or upgrade. After an upgrade, these tasks might have already been completed, but the related configurations should be confirmed.

- ["Configuring your external services" below](#)
- ["Disabling SSH and CLI" on the next page](#)
- ["Configuring Transport Layer Security \(TLS\)" on the next page](#)
- ["Trusting certificates for connections to your external services" on the next page](#)
- ["Importing x509v3 certificates as trusted root certificates" on page 10](#)
- ["Enabling FIPS mode" on page 11](#)
- ["Enabling Common Criteria mode" on page 12](#)
- ["Enabling Common Criteria mode for Samsung Knox devices" on page 12](#)
- ["Enabling Common Criteria mode for Samsung Knox devices" on page 12](#)
- ["Uploading a valid, trusted, FIPS-compatible certificate " on page 14](#)
- ["Selecting cipher suites" on page 15](#)
- ["Changing the port settings " on page 16](#)
- ["Configuring LDAP servers" on page 17](#)
- ["Changing the LDAP Server Sync Interval" on page 21](#)
- ["Setting the EULA or other login text" on page 26](#)

Configuring your external services

Configure your external services, such as:

- **Lightweight Directory Access Protocol** (LDAP) – See ["Configuring LDAP servers" on page 17](#).
- **Apple Push Notification Service** (APNS), – See "External and Internet rules" in the *On-Premise Installation Guide for Core and Enterprise Connector*.

- **Certificate providers** – See ["Trusting certificates for connections to your external services"](#) below, ["Importing x509v3 certificates as trusted root certificates"](#) on the next page, and ["Configuring Transport Layer Security \(TLS\)"](#) below.

Disabling SSH and CLI

To disable Secure Shell network protocol (SSH) and the command line interface (CLI), disable the option in the System Manager.

Procedure

1. In the Core System Manager portal, go to **Security > Settings > CLI**. The CLI Configuration page displays.
2. For SSH, select **Disable**.
3. Click **Apply**.

Configuring Transport Layer Security (TLS)

Configure Transport Layer Security (TLS) and trust certificates for connections to your external services before you enable FIPS mode and Common Criteria mode.

To configure TLS:

Procedure

1. In the Admin Portal, go to **Settings > System Settings > Security > TLS/SSL**.
2. Select **Allow only TLS/SSL connections certified by trusted CAs**.
3. Select **Verify revocation status**. However, note that some known services, such as Apple Push Notification Service (APNS), may have certificate revocation lists (CRLs) signed by root certificate authorities (CAs) that do not define key usage. Core will not fail these specific known outgoing TLS connections, but will log the behavior.
4. Click **Save**.

Trusting certificates for connections to your external services

When Transport Layer Security (TLS) is enabled, Core fails to connect to a service if it does not trust at least one Certificate Authority (CA) in the certificate chain that the service presents to Core. To trust the Root CA and required intermediate CA certificates, do the following:

Procedure

1. In the Admin Portal, go to **Services > Overview**.
2. Click **Verify All**.
3. Go to **Services > Trusted Root Certificates**.
(**Note:** this should be "Trusted CA certificates" but the UI label currently uses "Root.")
4. In the dropdown field, select **Show untrusted certs**.
5. Select the certificates that you want to trust.
6. Click **Actions > Trust**.

Importing x509v3 certificates as trusted root certificates

After doing the steps in "[Trusting certificates for connections to your external services](#)" on the previous page, you can import additional x509v3 certificates as trusted root certificates:

Procedure

1. In the Admin Portal, go to **Services > Trusted Root Certificates**.
2. Click **Add+** to open the **Upload Trust Root Certificate** dialog.
3. Add the certificate to the **File** box using the **Browse** button.
4. Click **Upload Certificate**.
5. Click **OK**.

The certificate is now in the list of trusted root certificates.

Deleting a trusted certificate

To delete a certificate:

Procedure

1. From the Admin Portal, go to **Services > Trusted Root Certificates**.
2. Click the drop-down box to the right, and select one of the following options:
 - **Show Trusted Certs**
 - **Show Untrusted Certs**
3. Select one or more certificates.
4. Select **Actions > Delete**.
5. Click **Yes** in the confirmation box.



You must restart the server to complete the process of deleting the selected trusted certificates.

Enabling FIPS mode

The Federal Information Processing Standards (FIPS) Publication 140-2 is a U.S. government computer security standard used to accredit cryptographic modules. FIPS 140-2 defines four levels of security, simply named **Level 1** to **Level 4**. It does not specify in detail what level of security is required by any particular application. MobileIron products are **FIPS 140-2 Level 1 Compliant**.



The following steps assume you are using a keyboard and monitor that are connected to the Core appliance, and that Core is running with the monitor displaying the Core CLI banner.

To enable FIPS:

Procedure

1. Enter **enable**.
2. When prompted, enter the enable secret you set when you installed Core.
3. Enter **configure terminal**.
4. Enter the following command to enable FIPS: **fips**
5. Enter the following command to proceed with the necessary reload: **do reload**.

```
System configuration may have been modified. Save? [yes/no]
```

6. Enter **yes**.

```
Configuration saved.  
Proceed with reload? [yes/no]
```


7. Enter **yes**.

```
Broadcast message from root (pts/0) (Sat Apr 16 21:54:52 2021):  
The system is going down for reboot NOW!
```

 The system will not be reachable until the reboot is complete.

8. Enter **show fips**. You should see a message that FIPS mode is enabled.

Enabling Common Criteria mode

 The following steps assume you are using a keyboard and monitor that are connected to the Core appliance, and that Core is running with the monitor displaying the Core CLI banner.

Complete the following steps to complete prerequisite setup for Common Criteria mode:

Procedure

1. Enter **enable**.
2. When prompted, enter the enable secret you set when you installed Core.
3. Enter **configure terminal**.
4. Enter **common_criteria_mode**.
5. Enter the following command to proceed with the necessary reload: **do reload**

The system will not be reachable until the reboot is complete.

6. Enter `show common_criteria_mode_status`.

You should see the following message: **Common Criteria Mode is enabled**.

Enabling Common Criteria mode for Samsung Knox devices

To enforce Common Criteria Mode on Samsung Knox devices, make sure you have configured a Samsung general policy with the Knox license associated with devices.

To enable Common Criteria mode for Samsung Knox devices:

Procedure

1. In the Admin Portal, go to **Policies & Configs > Policies**.
2. Select **Add New > Security**.
3. In the Android section, select **Common Criteria Mode (Samsung Knox Only and LG Only)**.



Although Core passes the setting to LG devices, Core supports Common Criteria mode only with Samsung Knox devices.

4. In the **Password** section:
 - a. **Password** field: Select **Mandatory**.
 - b. **Minimum Password Length**: Enter the required minimum.
 - c. **Minimum Number of Complex Characters**: Enter the required minimum.
 - d. **Maximum Password Age**: Enter the required maximum age of the password in days.
 - e. **Maximum Number of Failed Attempts**: Select **0**.
 - f. **Password History**: Select **0**.
5. Set **Device Encryption** (data-at-rest protection) to **On**.
6. Set **SD Card Encryption** (removable media data-at-rest protection) to **On**.
7. You can also select these other settings that support Common Criteria mode (**Bold** text indicates the field on the security policy):
 - a. Session locking policy (**Maximum Inactivity Timeout**)
 - b. Strict TLS requirements (**Require strict TLS for Apps@Work**)
8. Click **Save**.
9. Select **Actions > Apply To Label**.
10. Select a label that identifies the targeted Samsung Knox devices.
11. Click **Apply > OK**.

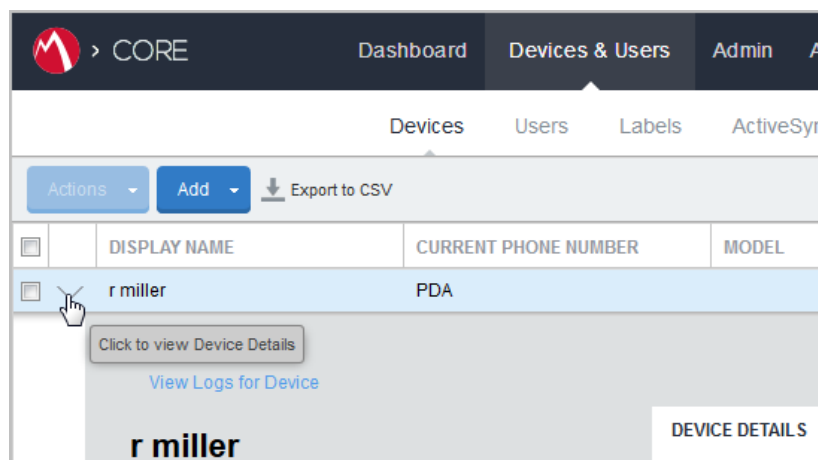
Confirming Common Criteria mode on Samsung Knox devices

To confirm that Common Criteria mode has been enabled for a Samsung Knox device:

Procedure

1. In the Admin Portal, go to **Devices & Users**.
2. In the **Devices** page, find the entry for the device.
3. Display device details (see next figure).

FIGURE 1. SELECTING DEVICE DETAILS



4. Click the **Policies** tab.
5. Scroll down to the **Common Criteria Mode** setting.
6. Once the entry for this setting displays **Enabled** in the **Device Value** column, Common Criteria Mode has been reported by the client as enabled.

Uploading a valid, trusted, FIPS-compatible certificate

You must upload a valid, trusted, FIPS-compatible Elliptic Curve Digital Signature Algorithm (ECDSA) certificate to Core.

Please note the following requirements for this action:

- It must be a publicly trusted certificate from a well-known Certificate Authority. You upload this certificate in the System Manager, in **Security > Certificate Mgmt.**
- On that screen, you upload this certificate for both the **Portal HTTPS** certificate and the **iOS Enrollment certificate.**
- When you use an ECDSA certificate and Core performs an Elliptic Curve Diffie-Hellman (ECDH) key agreement, the curve in the key agreement will match the curve in the certificate that you provide. Specifically:
 - If you upload a certificate that contains a key on the p-256 curve, Core performs its ECDH key generation on the p-256 curve.
 - If you upload a certificate with a key on the p-384 curve, Core performs its ECDH key generation on the p-384 curve.

For more information on creating this certificate, see ["Configuring an ECDSA client identity certificate for mutual authentication" on page 51.](#)

For more information about adding and removing keys and secrets, see the following sections:

- "Managing trusted certificates" in the *Getting Started with Core* guide.
- "Managing Wi-Fi-Settings" in the *Core Device Management Guide for Android and Android enterprise Devices* and *Core Device Management Guide for iOS and macOS Devices.*

Selecting cipher suites

When Core is in Common Criteria mode, the following restrictions apply when selecting cipher suites:

Selecting incoming cipher suites

The available cipher suites and the default list of selected cipher suites are displayed in the System Manager at **Security > Advanced > Incoming SSL Configuration.**

When Core is in Common Criteria mode, the System Manager displays all supported incoming cipher suites to Core from external servers. The following evaluated cipher suites (and the RFCs in which they are defined) are the only ones allowed while operating in an evaluated configuration:

- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289



The administrator must configure the incoming SSL configuration to include only these four cipher suites, or a subset.

Selecting outgoing cipher suites

The available cipher suites and the default list of selected cipher suites are displayed in the System Manager at **Security > Advanced > Outgoing SSL Configuration**.

The following eight cipher suites were evaluated and should be used while operating in an evaluated configuration:

- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289,
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

Changing the port settings

To change the port settings:

Procedure

1. In the System Manager, go to **Settings > Port Settings**.
2. Set **Sync Service Port** to **Disable**.
3. Set **Provision Protocol** to **https**.

All remote administration access to the Admin Portal and the System Manager must be performed through a HTTPS/TLS protected connection. Do not allow non-secure protocols such as HTTP.

4. Click **Apply**.

Configuring LDAP servers

When configured, Core can interact with LDAP servers. You can configure multiple LDAP servers, but each server must contain unique configuration.



The Core Connector does not support certificate-based authentication. This means that once you enable Connector service, the "Upload X509 Certificate" option in LDAP preferences is not available.

Procedure

1. From the Admin Portal, go to **Services > LDAP**.
2. Click **Add New** to open the New LDAP Setting page.
3. Edit the fields as necessary.

Refer to the "[LDAP Server window](#)" on page 19 table for details.

4. Scroll to the **LDAP Groups** setting to specify the set of LDAP groups that Core gets from the LDAP server. Only these groups are available throughout the Admin Portal for viewing or selection.
5. Go to **Search By LDAP Groups**, enter the first characters of an LDAP Group that you want to select.
6. Click the search icon.

The LDAP Groups in the LDAP server that match the search request appear in the **Available** section.

7. Click the right arrow to move one or more LDAP groups to the **Selected** section.
8. Repeat steps 6 through 8 for other LDAP Groups.
9. Click **Advance Options** to configure LDAP v3 properties.



Configurations in the **Advanced Options** pane apply only to LDAP v3 servers.

10. Select the authentication method between the client and server used in the SASL exchange.
 - **Bind** (default): This method uses the directory DN for authentication.
 - **Kerberos v5** (SASL): This method uses mutual authentication.

11. Select the user ID format from the **Authentication User ID Format** drop-down list.
 - User Principal
 - User UPN
 - User DN
 - User DN with RFC2829 prefix
 - User Principal with RFC2829 prefix
12. Select the group member format from the **Group Member Format** drop-down list.
 - DN
 - UID
13. Select the parameter for negotiating the authentication from the **Quality of Protection** drop-down list.



LDAP v3 supports the Quality of Protection feature, which is not an LDAP v2-supported feature.

- **Authentication only** is used for authenticating a user to a server.
 - **Authentication with integrity protection** is used to ensure that subsequent LDAP requests and responses are protected against tampering.
 - **Authentication with integrity and privacy protection** is used to ensure that subsequent LDAP requests and responses are encrypted and therefore protected against unintended monitoring. Privacy protection automatically entails integrity protection.
14. Select the LDAP authentication method.
 - **Use Client TLS Certificate:** Select this to use the X509 certificate for authentication.
 - Go to **Services > LDAP > Preferences** to upload the client X509 certificate that Core presents to the LDAP server
 - **Request Mutual Authentication:** Select this to verify both the identity of the user that is requesting authentication as well as server providing the requested authentication.
 15. Select **Enable Detailed Debug** to enable Java Naming and Directory Interface (JNDI) debugging for LDAP communication.
 16. Enter additional (and optional) properties in the **Additional JNDI Context Properties** field.

17. Most environment properties are predefined but some, such as *language*, *security.credentials*, *security.principle*, are implementation-specific. Properties defined here replace any values that are previously defined, and will take effect the next time the property is invoked. If a context does not have a particular environment property, it behaves as if it has that environment property with its default value. For example,
 - To set the language for Japanese, enter `Context.LANGUAGE, "ja-JP"`
 - To set the credentials to the string "secret", enter `Context.SECURITY_CREDENTIALS, "secret"`
 - To set the principal name to the distinguished name "cn=admin, o=MI, c=us," enter `Context.SECURITY_PRINCIPAL, "cn=admin, o=MI, c=us"`
18. Click **View LDAP Browser** to view the LDAP server directory tree structure.
19. Click **Test** to open the **LDAP Test** window
20. Enter user or group identifier in the appropriate field.
21. Click **Submit**. A result page displays if the user was configured on the LDAP server.
22. Return to the LDAP page and click **Save**. A dialog appears informing of traffic disruption and asks to proceed.
23. Click **Yes**. A dialog appears informing the status.
24. Click **OK**. The server you created appears on the LDAP page.



LDAP Server window

The following table summarizes fields and descriptions in the **LDAP Server** window:

TABLE 1. LDAP SERVER FIELDS

Fields	Description
Directory URL	Enter the URL to the LDAP server. Make sure to start with "ldap://" or "ldaps://". When using "ldaps://" (LDAP over SSL) : <ul style="list-style-type: none"> • You need an X509 certificate for LDAP authentication.

TABLE 1. LDAP SERVER FIELDS (CONT.)

Fields	Description
	<ul style="list-style-type: none"> • The following fields of the certificate presented by the LDAPS server to Core must match the URL: <ul style="list-style-type: none"> ◦ the Common Name (CN) ◦ the Subject Alternative Name (SAN) ◦ the DNS name • If no match exists, the connection request fails. <hr/> <p> If the certificate has a SAN field, Core ignores the CN value and seeks a match in the SAN list. Using the CN field is deprecated. Therefore, Core checks the CN only if the SAN is not present.</p> <hr/> <p>You do not need to specify the ports when you use these default ports: 389 (LDAP) or 636 (LDAPS).</p>
Directory Failover URL	Enter a secondary URL, if available.
Directory UserID	Enter the primary user ID, for example, userid@local.domain. Make sure to include the domain, e.g., @local.domain, with the user ID.
Directory Password	Enter the password for the user ID set above.
Search Results Timeout	Do not change default of 30 seconds unless you get connection errors.
Chase Referrals	<p>Select Enable if you are using a multi-forested domain. This indicates you want to use alternate domain controllers when the targeted domain controller does not have a copy of the requested object.</p> <p>Select Disable if you do not use alternate domain controllers.</p> <hr/> <p> Enabling the Chase Referrals option delays LDAP authentication.</p>
Admin State	Select Enable to put the server to service. Make sure to enable the Admin state or the LDAP server will be invisible.
Directory Type	<p>Select Domino for the IBM Lotus Domino server platform. The default DN and other LDAP search filters are automatically changed to the Domino server.</p> <p>Select Active Directory for the Microsoft Windows server platform.</p>
Domain	Enter the domain name for the Active Directory. This information will automatically traverse all levels of the tree and use to populate Base DN, parent entry.

Changing the LDAP Server Sync Interval

The default interval for synchronization between Core and the LDAP server is 24 hours. You can change this interval for all configured LDAP servers. You might want to change the interval to ensure updated information when the LDAP server data is changing frequently.



For LDAP groups, each synchronization syncs only the LDAP groups that you specified in the LDAP Setting page for each LDAP server at **Services > LDAP**.

To change the LDAP sync interval:

Procedure

1. From the Admin Portal, go to **Services > LDAP > Preferences**.
2. Select the preferred interval from the drop-down.

Intervals range from 15 minutes to 24 hours.
3. Click **Save**.

Configuring mutual authentication

Core supports mutual authentication, which means that not only must the device trust Core, but Core must trust the device. Therefore, with mutual authentication, a registered device can continue to communicate with Core only if the device provides the right certificate to Core. Mutually authenticated communication between the device and Core enhances security. A device authenticating to Core with a certificate is also known as certificate-based authentication to Core.

As part of configuring mutual authentication, you select a certificate enrollment setting that specifies how the identity certificate that the device presents to Core is generated.



For devices, the client identity certificate that the device presents to Core must be an ECDSA (Elliptical Curve Digital Signature Algorithm) certificate.

The setting on MobileIron Core to enable mutual authentication is in the Admin Portal in **Settings > System Settings > Security > Certificate Authentication**.



Important: Once mutual authentication is enabled on Core, it cannot be disabled.

When devices use mutual authentication

The following table summarizes when devices use mutual authentication and the port they use in communication with Core.

TABLE 2. CORE MUTUAL AUTHENTICATION (MA) SETTING IMPACT TO DEVICE COMMUNICATION

	New Core installation or Core upgrade in which: MA setting was NOT enabled before upgrade	New Core installation in which: You enable MA setting after installation or Core upgrade in which: MA setting was enabled AFTER the upgrade	Core upgrade in which: MA setting was enabled BEFORE the upgrade
Mutual authentication setting	Not enabled	Enabled	Enabled
Device client			
Android: Mobile@Work (all Mobile@Work versions that Core supports)	Port: 9997 MA: not used	Devices that register after enabling MA: <ul style="list-style-type: none"> • Port: 443 • MA: used Devices that were already registered: <ul style="list-style-type: none"> • Port: 9997 • MA: not used. 	Port: 443 MA: used
iOS: iOS MDM check-in	Port: 443 MA: not used	Port: 443 MA: used	Port: 443 MA: used.

Mutual authentication identity certificate for Core

You provide an identity certificate for Core to use in mutual authentication in the Portal HTTPS certificate. You configure this certificate on the System Manager at **Security > Certificate Mgmt**. The certificate is the identify certificate and its certificate chain, including the private key, that identifies Core, allowing the devices to trust Core. This certificate must be a publicly trusted certificate from a well-known Certificate Authority when using mutual authentication.

Mutual authentication client identity certificate

You enable mutual authentication for iOS and Android devices in the Admin Portal in **Settings > System Settings > Security > Certificate Authentication**. The certificate enrollment setting specifies how the identity certificate that the device will present to Core is generated.

By default, the certificate enrollment setting for mutual authentication is generated with Core as a local Certificate Authority (CA). Most customers use the default selection. However, if necessary due to your security requirements, you can instead specify a certificate enrollment setting that you create.



IMPORTANT: If you use a certificate enrollment setting for mutual authentication, you cannot use it for any other purpose. For example, you cannot use it in VPN or Wi-Fi configurations.

If you use a certificate enrollment setting that uses an intermediate CA, make sure that all the intermediate CA certificates and the root CA certificate are included in Core's trusted root certificates.

Supported custom attributes for mutual authentication certificates

Certificate subject of the client device identity certificate must be in the following format:

CN= <random value>, (Preferred Distinguished Name)

where *random value* must be provided by one of these customer attributes: **\$RANDOM_32\$**, **\$RANDOM_64\$**, **\$TIMESTAMP_MS\$**.

\$RANDOM_32\$ is the default, and provides a 32-bit random value.



TIP: The larger random values are less likely to have collisions, and a time stamp value is the most likely to be unique.

Example

CN=\$RANDOM_32\$,O=ARMY,OU=RANGERS

Handling client identity certificate expiration for Android devices

Mobile@Work for Android handles the expiration of the client identity certificate used for mutual authentication between Mobile@Work for Android and Core. In the Admin Portal, on the sync policy for the device, specify a renewal window for the certificate. The renewal window is a number of days prior to the certificate expiration. When Mobile@Work determines the renewal window has begun, it requests a new certificate from Core.

- If Mobile@Work is out of contact with Core during the renewal window, but is in contact again within 30 days after the expiration, Mobile@Work requests a new certificate from Core.
- If Mobile@Work is not in contact with Core either during the renewal window or within 30 days after the expiration, the device will be retired and will need to re-register with Core.

Mobile@Work versions prior to 10.1 do not support certificate expiration. When the certificate expires, the device user must re-register Mobile@Work.

Procedure

1. In the Admin Portal, go to **Policies & Configs > Policies**.
2. Select the appropriate sync policy.
3. For **Mutual Certificate Authentication Renewal** window, enter the number of days prior to the expiration date that you want to allow devices to renew their identity certificate. Enter a value between 1 and 60. A blank value defaults to 60 days.
4. Click **Save**.
5. Click **OK**.

Client identity certificate expiration for iOS devices

Mobile@Work for iOS handles the expiration of the client identity certificate used for mutual authentication between Mobile@Work for iOS and Core. In the Admin Portal, on the sync policy for the device, specify a renewal window for the certificate. The renewal window is a number of days prior to the certificate expiration. When Mobile@Work determines the renewal window has begun, it requests a new certificate from Core.

- If Mobile@Work is out of contact with Core during the renewal window, but is in contact again within 30 days after the expiration, Mobile@Work requests a new certificate from Core.
- If Mobile@Work is not in contact with Core either during the renewal window or within 30 days after the expiration, the device will be retired and will need to re-register with Core.

Mobile@Work versions prior to 11.1.0 do not support certificate expiration. When the certificate expires, the device user must re-register Mobile@Work.

Procedure

1. In the Admin Portal, go to **Policies & Configs > Policies**.
2. Select the appropriate sync policy.
3. For **Mutual Certificate Authentication Renewal** window, enter the number of days prior to the expiration date that you want to allow devices to renew their identity certificate. Enter a value between 1 and 60. A blank value defaults to 60 days.
4. Click **Save**.
5. Click **OK**.

Enabling mutual authentication for Apple and Android devices

As discussed in [Configuring mutual authentication](#), create a certificate enrollment setting if you do not want to use the default local certificate enrollment setting for mutual authentication. The certificate enrollment setting must be the **Decentralized** option.



When you enable mutual authentication, change the certificate enrollment selection for mutual authentication *before* any more devices register. Any devices already registered and using mutual authentication will not be able to check-in with Core. Those devices will need to re-register with Core. Note that devices already registered but not using mutual authentication can continue to check-in.

If you are using iOS devices with the Apps@Work web clip using certificate authentication, change the Apps@Work Port field in the System Manager in **Settings > Port Settings**. Ivanti recommends port 7443. However, you can use any port except the port that the Admin Portal uses, which is either 443 or 8443.

Procedure

1. In the Admin Portal, go to **Settings > System Settings > Security > Certificate Authentication**.
2. Select **Enable client mutual certification** on Android client, iOS client and Apple MDM communication.
3. In the **Certificate Enrollment Configuration** field, most customers use the default selection. Otherwise, select a certificate enrollment setting.
4. Click **Save**.

Setting the EULA or other login text

You can configure Core to display an End User License Agreement (EULA) or any other text on the following user interfaces:

- Admin Portal login screen
- System Manager login screen
- a CLI session
- self-service user portal login screen

Procedure

1. In the Admin Portal, go to **Settings > System Settings > General > Login**.
2. Select **Enable Login Text Box**.
3. In **Text To Display**, enter the text.



Core treats the text as plain characters. It does not recognize, for example, HTML tags. The text must be ASCII only; no multi-byte characters are allowed.

4. Click **Save**.

The Admin Portal and the System Manager display this text the next time a user logs in.



The Core CLI command banner, available in CLI CONFIG mode, also sets this text.

Disabling login text

To disable this setting:

1. In the Admin Portal, go to **Settings > System Settings > General > Login**.
2. Uncheck **Enable Login Text Box**.

The text you had entered in **Text To Display** is grayed out.

3. Click **Save**.

The login screens and CLI session do not show the text the next time a user logs in.

Setting up Core with a closed network / AOSP deployment

There are situations where the onboarding, registration and management of devices is limited and requires a different approach. Examples of these kinds of situations are:

- In an environment that does not have connectivity to Google mobile services (GMS) due to restrictions in the organization or due to a closed network.
- Where devices that do not have Google mobile services but vendors have enabled Android Enterprise AOSP (Android Open Source Project.)
- Integrated deployment (GMS/Non-GMS) - the entire Core instance serves devices in full Android Enterprise mode (for example, Samsung devices) and also devices that do not have GMS (for example, AR/VR devices.)

This feature applies to Android 6 and supported newer versions.

Use cases for AOSP deployments

AOSP deployment can work side-by-side with Android Enterprise deployment. Supported modes of registration and use cases include:

- Device Admin mode
- Device Owner mode

This feature supports the following use cases:

TABLE 3. USE CASES FOR CLOSED NETWORK / AOSP DEPLOYMENT

Use Case	Expected Result
Only the AOSP configuration is pushed	AOSP in Device Admin or Device Owner mode.
Both AOSP and Android Enterprise configurations are pushed	Android Enterprise configuration takes priority over AOSP.
Device is registered in AOSP and the Android Enterprise configuration is pushed to the registered device	The registered device retains AOSP mode. Android Enterprise configuration applies only to fresh registrations.

Complete the following tasks to enable a closed network / AOSP deployment:

1. ["Enabling a closed network / AOSP deployment in Core" below.](#)
2. ["Creating a new Android Enterprise configuration" on page 58.](#)
3. ["Provisioning the Android device" on page 31.](#)
4. ["Managing the closed network / AOSP devices" on page 32.](#)

Enabling a closed network / AOSP deployment in Core

Administrators have the ability to enable a closed network / AOSP deployment in Core whether Android Enterprise is already enabled or not. Once the closed network / AOSP is deployed, Administrators can optionally switch off Android Enterprise on the instance.

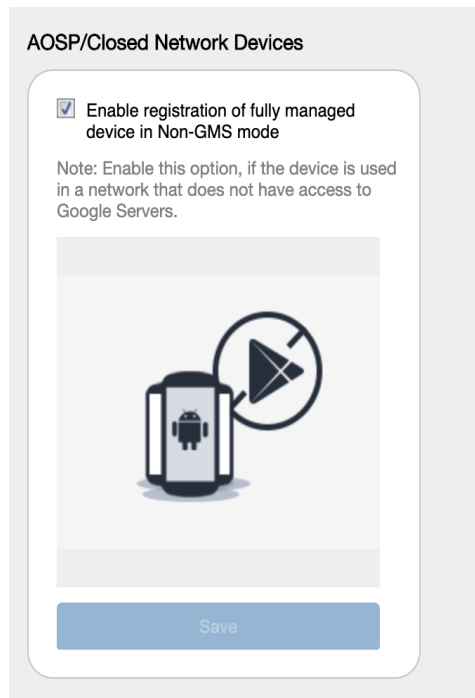


Core does not impose any special restrictions for Google calls, either from the client or the server. The device user decides how to handle public apps pushed from Core when the device is registered in **Work Managed Device - Non GMS** mode.

Procedure

1. In Core, go to **Services > Google**.
2. Scroll down and find the **AOSP/Closed Network Devices** tile.

FIGURE 1 . AOSP/CLOSED NETWORK DEVICES TILE



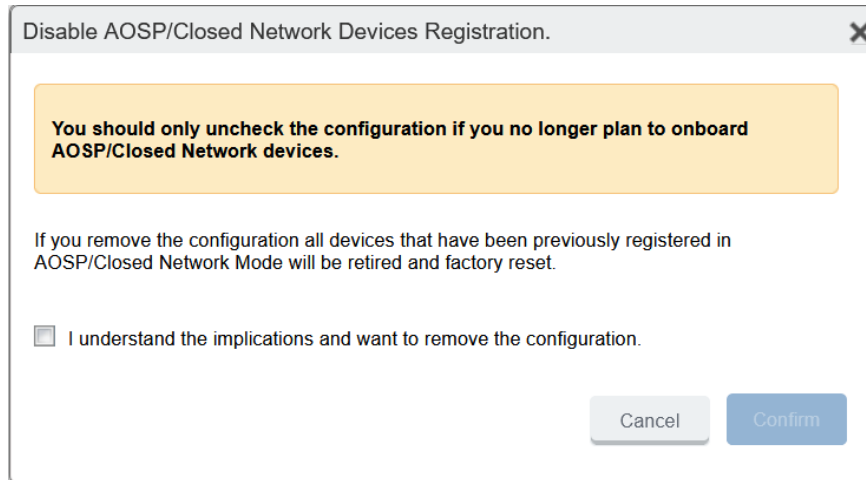
3. Select the **Enable registration of fully managed device in Non-GMS mode** check box and then click **Save**.

Disabling AOSP / closed network deployment

You can disable the AOSP / closed network deployment. Disabling AOSP in the Google Services page will cause the registered devices in Device Owner (DO) mode to retire. Devices in Device Admin (DA) mode will stay intact.

1. In the **AOSP/Closed Network Devices** tile, de-select the check box and then click **Save**. The Disable AOSP/Closed Network Devices Registration dialog box opens.

FIGURE 2. DISABLE AOSP/CLOSED NETWORK DEVICES REGISTRATION WINDOW



2. Select the **I understand the implications and want to remove the configuration** check box.
3. Click **Confirm**.

Creating a new Android Enterprise configuration

Administrators will need to create a new Android Enterprise configuration for the closed network or for an AOSP deployment, and then add the new configuration to a label.

Procedure

1. Go to **Policies & Configs > Configs**.
2. Click **Add New > Android Enterprise Setting**. The New Android Enterprise (all modes) Setting dialog box opens.
3. Enter a **Name** and **Description**.
4. Select the **Enable Closed Network/AOSP deployment** check box. All other options in the dialog box become hidden except **Enable Runtime Permissions**.
5. Click **Save**.
6. Apply a label to the configuration.



If two Android Enterprise configurations (one with Android for Work and one with AOSP) are pushed to the same device, the Android for Work configuration takes priority.



As Firebase Cloud Messaging (FCM) services will not be available, Ivanti recommends administrators to modify the **Sync Policy > Sync Interval** field to 30 minutes or to the lowest value as needed by the administrator.



Once the **Enable Closed Network/AOSP deployment** check box in the Android Enterprise Setting (configuration) dialog box is enabled, after saving, the option will be grayed out. There is no option for the administrator to disable it. As a workaround, the administrator should either remove the label or delete the configuration.

Provisioning the Android device

For a closed network / AOSP deployment, only Android Enterprise devices in Work Managed Device mode can be provisioned. In order to provision the device, the administrator needs to download the Device Policy Controller (DPC.)

Provisioner app can be used to install an MDMPP client hosted on any local HTTP server. Provisioner app can be found on Google Play:

<https://play.google.com/store/apps/details?id=com.mobileiron.client.android.nfcprovisioner>

What the Administrator does

Procedure

1. Add the URL for the Internet Information Services (IIS) server (where the Device Policy Controller (DPC) is hosted) into the Provisioner app. The DPC generates a QR code.
2. The Android device is rebooted to factory settings. Device registration is complete.

What the device user does

The following steps complete the device registration process for Android devices in Google Mobile Services (GMS) and non-GMS environments.

Provisioning non-GMS devices when ADB access is enabled on a device

You can provision non-GMS devices when Android debug bridge (ADB) access is enabled.

Procedure

1. Sideload and install the Mobile@Work app onto the device.
2. Run the below command to provision the device into Device Owner (DO) mode. Be sure that the Mobile@Work app is installed before running the command.

```
adb shell dpm set-device-owner com.mobileiron.mdmp/.receiver.MIDeviceAdmin
```

3. Register the device by entering the server URL, username, and password. Device registration is complete.

Provisioning Non-GMS devices using the Provisioner app

The Provisioner app can be used to install an MDMPP client hosted on any local HTTP server.

1. Select the **MDMPP** client variant listed on the Provisioner app screen. Selecting **MDMPP** enables the following options:
 - a. You may specify a URL in a QR code to a client location on an HTTP server.
 - b. You may specify a URL to a local public key infrastructure (PKI) file, if one is needed to trust the server's certificate in an air-gap environment.

Provisioning GMS devices

Procedure

1. The device user scans the QR code. The DPC is downloaded onto the device.
2. The Mobile@Work client opens and requests the device user to enter the server URL. (The server URL and username can be provisioned as part of the QR code.)
3. The device user enters the server URL, username and password. Device registration is complete.

Managing the closed network / AOSP devices

Registration status values

Upon registration to Core, the device sends all device details to Core. The **Device Details page > Registration Status** field lists the following values:

TABLE 4. REGISTRATION STATUS VALUES

Action	Registration Status value
Android Enterprise configuration sent to device	Work Managed Device
Closed network / AOSP configuration sent to device	Work Managed Device - Non GMS
Device does not receive the AOSP configuration	The device is retired (factory reset.)

Closed network / AOSP device capabilities

After successful registration, devices will be able to receive and provision the following:

TABLE 5. CLOSED NETWORK / AOSP DEVICE CAPABILITIES

Type	Description
Configurations	<ul style="list-style-type: none"> • Android Enterprise • Android XML • Android APN • Exchange • MTD • Certificates • VPN • Wi-Fi
Policies	<ul style="list-style-type: none"> • Android Quick Setup • Android Kiosk • Firmware policy • Security • Privacy • Lockdown • MTD local actions • MTD anti-phishing • Compliance policy • Sync policy
App Management	<ul style="list-style-type: none"> • Support for in-house apps with configurations/restrictions • Apps@Work
Standard device management capabilities	All the supported device management commands of Android Enterprise work for closed network / AOSP deployment, except "Shared Kiosk- Signout."

App Management

- With a closed network / AOSP deployment, devices registered as a non-GMS device will have access to all in-house applications through Apps@Work.
- In non-closed networks / AOSP deployments, all apps need to be uploaded as in-house apps using their .apks since there is no access to Google's application bundles.
- When applying app restrictions, make sure to have the **Install this app for Android enterprise** and **Enable AOSP app restrictions** check boxes selected.

General Core configuration tasks

These tasks relate to configuring Core and using its capabilities for activities that are not specific to Android or iOS devices.

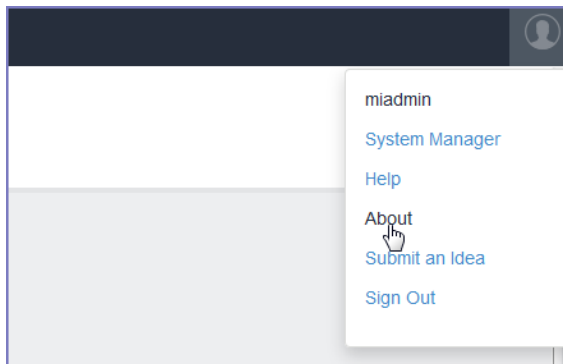
Determining your Core version	35
Configuring administrators for roles defined by federal requirements	36
Configuring the Admin Portal timeout	41
Configuring certificate authentication for password-less access	42
Setting up PIN registration	42
Configuring the sync interval for a device	42
Limiting the number of devices that users can enroll	43
Checking registration and check-in/connectivity status	43
Querying hardware and software information about a device	44
Querying the installed apps on a device	44
Installing policies on a device	44
Using labels for application groups	45
Configuring an ECDSA client identity certificate for mutual authentication	51
Updating Core	53

Determining your Core version

To determine the version of Core you are currently running:

Procedure

1. In the Admin Portal, expand the user menu on the far right.



2. Select **About**.
The displayed About box lists the version of the Core software.

Configuring administrators for roles defined by federal requirements

You can configure administrators with one or more of the following roles:

- **Administrator** or **Server primary administrator** – The user ID that you created when installing Core. This user can access both the Admin portal and the System Manager. In the Admin portal, this user is automatically configured to be a *Core super administrator*. A super administrator is assigned to the Core device space called the *global space* and can perform all administrative actions.
- **Mobile device user** – This is a user added in the Admin portal, but not assigned an administrative role.
- **Enrolled mobile device** – This is a mobile device that has been enrolled by a mobile device user.
- **Application access group** – The target of evaluation (TOE) allows “Labels” to be defined and apps to be assigned. Devices can also be assigned to labels and the associated services to restrict or allow access to corresponding apps.
- **Security configuration administrator** – This is a user with access to the web-based System Manager and console command-line interfaces, as well as most of the Admin portal manage roles. The security configuration administrator is responsible for:
 - Security configuration of the server
 - Setting up and maintenance of mobile device security policies
 - Defining device user groups
 - Setup and maintenance of device user group administrator accounts
 - Defining privileges of device user group administrators.

On Core, these capabilities include can include System Manager tasks, as well as Admin portal tasks such as managing users, administrators, configurations, policies, settings, services, apps, and labels. See ["Creating a security configuration administrator" on page 38](#).

- **Device user group administrator** – This is a user that minimally has the **Manage User Admin Portal** role, but is not assigned the Manage Administrators and Device Spaces role. See ["Creating a device user group administrator" on page 40](#).
- **Auditor** – This is a user that minimally has the **Manage Logs and Events** Admin portal role. If necessary for a given deployment, an auditor can be provided access to the System Manager portal to have access to low-level protocol audit records (application logs). See ["Creating an auditor administrator" on page 40](#).

Creating more than one server primary administrator

To create another server primary administrator:

Procedure

1. Log into the System Manager using the user ID that you created when installing Core.
2. Go to **Security > Identity Source > Local Users**.
3. Click **Add** to open the **Add New User** window.
4. Modify one or more of the fields, as necessary. Refer to the [Add New User window](#) table for details.
5. Click **Apply > OK**.

Add New User window

The following table summarizes fields and descriptions in the **Add New Users** window:

TABLE 6. ADD NEW USER FIELDS

Fields	Description
User ID	Enter the unique identifier to assign to this user. The user ID is case sensitive.
First Name	Enter the user's first name.
Last Name	Enter the user's last name.
Password	<p>Enter a password for the user.</p> <p>Valid passwords are determined by the password policy for System Manager local users.</p> <p>Enter a password for the user based on the Password Policy configured by the administrator in the System Manager (Security > Identity Source > Password Policy). However the following password requirements cannot be changed:</p> <ul style="list-style-type: none">• cannot be the same as the user ID• cannot contain the Grave accent character• cannot contain the space character• cannot have 4 or more repeating characters• users cannot change a password more than once during a 24 hour period
Confirm Password	Confirm the password for the user.

TABLE 6. ADD NEW USER FIELDS (CONT.)

Fields	Description
Space	This field is not configurable. It is set to the global space.
Email	Enter the user's email address.
EDIPI	Department of Defense customers only: Enter the user's the Department of Defense identification number, also known as the Electronic Data Interchange Personal Identifier. This field is required if your configuration on Security > Advanced > Portal Authentication specifies certificate authentication for access to the System Manager using a common access card (CAC).

Creating a security configuration administrator

You can create a security configuration administrator to have access to:

- Both the System Manager and the Admin Portal – See "[Log access to System Manager and Admin portal](#) " below.
- Only the Admin Portal – See "[Log access to the Admin portal only](#)" on the next page.

Log access to System Manager and Admin portal

Using a system primary administrator ID, complete the following steps:

Procedure

1. Create a security configuration administrator:
 - a. In the System Manager, go to **Security > Identity Source > Local Users**.
 - b. Create a new admin user.
2. In the Admin Portal, go to **Admin > Admins**.
3. Select the new user.
4. Click **Actions > Edit Roles**.
5. Select the following roles, and deselect other roles:

- a. In the **Label Management** section, select **Manage label**.
 - b. In the **User Management** section, select **View user**.
 - c. In the **App Management** section, **Manage app**.
 - d. In the **Configuration Management** section, **Manage configuration**.
 - e. In the **Policy Management** section, **Manage policy**.
 - f. In the **Settings and Services Management** section, **Manage settings and services**.
 - g. In the **Admin Management** section, **Manage administrators and device spaces**.
6. Click **Save**.

Log access to the Admin portal only

Using a system primary administrator ID, complete the following steps:

Procedure

1. Go to the Admin Portal to **Devices & Users > Users**.
 - a. Add a local user that is a security configuration administrator.
2. After creating the local user, go to **Admin > Admins**.
3. Select the new user.
4. Click **Actions > Assign To Space**.
 - a. In **Select Space**, select **Global**. Optionally, you can select a different space.
 - b. Scroll down to the **Label Management** section and select **Manage label**.
 - c. Scroll down to the **User Management** section and select **View user**.
 - d. Scroll down to the **App Management** section and select **Manage app**.
 - e. Scroll down to the **Configuration Management** section and select **Manage configuration**.
 - f. Scroll down to the **Policy Management** section and select **Manage policy**.
 - g. Scroll down to the **Settings and Services Management** section and select **Manage settings and services**.
 - h. Scroll down to the **Admin Management** section and select **Manage administrators and device spaces**.
5. Click **Save**.

Creating a device user group administrator

You can create an administrator to manage device user groups. Using a system primary administrator ID, complete the following steps:

Procedure

1. Go to the Admin Portal to **Devices & Users > Users**.
2. Add a local user that is a security configuration administrator.
3. After creating the local user, go to **Admin > Admins**.
4. Select the new user.
5. Click **Actions > Assign To Space**.
6. In **Select Space**, select **Global**. Alternately, you can select a different space.
 - a. Scroll down to the **User Management** section and select **Manage user**.
 - b. Scroll down to the **Device Management** section and select:
 - a. **Manage devices**
 - b. **Wipe device**
 - c. **Add device**
 - d. **Manage ActiveSync device**
 - e. **Delete retired devices**
7. Click **Save**.

Creating an auditor administrator

You can configure an auditor to have access to log into:

- Both the System Manager and the Admin portal – See "[Log access to System Manager and Admin portal](#)" below
- The Admin portal only – "[Log access to the Admin portal only](#)" on the next page

Log access to System Manager and Admin portal

Using a server primary administrator ID, complete the following steps:

Procedure

1. In the System Manager, go to **Security > Identity Source > Local Users**.
2. Add a local System Manager user that is a security configuration administrator.
3. In the Admin Portal, go to **Admin > Admins**.
4. Select the new user.
5. Click **Actions > Edit Roles**.
6. Deselect each role.
7. Scroll down to the **Logs and Events Management** section and select **Manage events**.
8. Click **Save**.

Log access to the Admin portal only

Using a server primary administrator ID, complete the following steps:

Procedure

1. Go to the Admin Portal to **Devices & Users > Users**.
2. Add a local user that is a security configuration administrator.
3. After creating the local user, go to **Admin > Admins**.
4. Select the new user.
5. Click **Actions > Assign To Space**.
6. In **Select Space**, select **Global**.
Alternately, you can select a different space.
7. Scroll down to the **Logs and Events Management** section and select **Manage events**.
8. Click **Save**.

Configuring the Admin Portal timeout

By default, the Admin Portal session ends when no activity is detected for 60 minutes. To change this interval:

Procedure

1. In the Admin Portal, go to **Settings > System Settings > General > Timeout**.
2. Select the preferred timeout interval from the drop-down list.
3. Click **Save**.

Configuring certificate authentication for password-less access

To set up certificate authentication to the Core user portal and Admin Portal, see the following sections of the *Core System Manager Guide*:

- "Certificates required for certificate authentication to the user portal or Admin Portal"
- "Configuring certificate authentication to the user portal"
- "Configuring certificate authentication to the Admin Portal"

Setting up PIN registration

You can configure Core so that when users register an Android or iOS device to Core from Mobile@Work, they enter a PIN instead of a password.

Configuring the sync interval for a device

Mobile devices periodically synchronize with a Core MDM server during check-in. The mobile device and Core synchronize profiles, configurations, and app inventory. This synchronization interval is configurable, and supports compliance reporting.

To configure the frequency for starting the synchronization process between a device and Core:

Procedure

1. In the Admin Portal, go to **Policies & Config > Policies**.
2. Select a sync policy.
3. Set **Sync Interval** to the number of minutes between synchronizations.
4. Click **Save**.

For information on creating a sync policy and applying it to devices using labels, see "[Installing policies on a device](#)" on page 44.

Limiting the number of devices that users can enroll

Use this procedure to restrict the number of devices users can register at a time or remove all restrictions.

Procedure

1. From the Admin Portal, go to **Settings > System Settings > Users & Devices > Registration**.
2. In **Per-User Device Limit**, enter the number of devices each user can register.
 - a. Specify a limit from 1 to 50 devices.
 - b. Leave the box blank, the default value, to indicate no limit.
3. Click **Save**.

Checking registration and check-in/connectivity status

Use the Admin Portal to determine whether a device has completed registration and has checked in recently. This same procedure indicates the connectivity status of the device.

Procedure

1. In the Admin Portal, go to **Devices & Users > Devices**.
2. Locate the entry for the device in question.

FIGURE 1. CHECKING REGISTRATION AND CONNECTIVITY FOR A DEVICE

	DISPLAY NAME	CURRENT PHONE NUMBER	MODEL	MANUFACTURER	PLATFORM NA...	HOME COUNTRY NAME	STATUS	REGISTRATION DATE	REGISTRATION ST...	LAST CHECK-IN	OWNER	OPERATO
^	a		Lumia 630 Du...	NOKIA	Windows Phon...		Active	2015-11-13 10:14:3...		1 d 19h	Company	
^	a	PDA 4	VMware Virtual ...	VMware, Inc.	Windows 10		Active	2015-11-13 02:16:4...		2 d 1h	Company	

3. In the **Status** column, confirm that the device is **Active**. Active means that the device is properly registered.
4. Check the **Last Check-In** column to determine the elapsed time since the device last contacted the server.

Querying hardware and software information about a device

To view the hardware and software information of a device:

Procedure

1. In the Admin Portal, go to **Devices & Users > Devices**.
2. Find the row for the device. You can scroll down for the device, or use the label filter or search filter.
3. View the following columns:
 - **Model**
 - **Manufacturer**
 - **PlatformName**
4. Click on the up arrow to the left of the display name for the device.
5. The device details panel displays. More software and hardware information displays about the device in the **Device Details** tab. For more information, see "Displaying device assets" in *Getting Started with Core*.

Querying the installed apps on a device

To view the apps that are installed on a device:

Procedure

1. In the Admin Portal, go to **Devices & Users > Devices**.
2. Find the row for the device. You can scroll down for the device, or use the label filter or search filter.
3. Click on the up arrow to the left of the display name for the device. The device details panel displays.
4. Click the **Apps** tab. The list of apps installed on the device displays. For more information, see "Displaying more device and user information" in *Getting Started with Core*.

Installing policies on a device

To install policies on a device, see the following sections:

- ["Enabling Common Criteria mode for Samsung Knox devices" on page 12](#)
- ["Configuring mutual authentication" on page 21](#)
- ["Configuring the sync interval for a device" on page 42](#)

- "Configuring an ECDSA client identity certificate for mutual authentication" on page 51
- "Creating a new Android Enterprise configuration" on page 58
- "Closed network / AOSP device capabilities" on page 33
- "Creating a new Android Enterprise configuration" on page 58
- "Configuring the Android warning banner " on page 75
- "Configuring allowed app sources for Android Samsung Knox devices" on page 76
- "Wi-Fi settings for Android devices" on page 79
- "Restricting Wi-Fi access to specific networks on Android devices" on page 82
- "Disabling the Developer options menu on Android devices" on page 85
- "Prohibiting device users from unenrolling" on page 86
- "Disabling biometric authentication on Android devices" on page 86
- "Setting up the security policy to trigger the compliance action when the violation occurs" on page 88
- "Wi-Fi settings for iOS devices" on page 91
- "Whitelisting Wi-Fi networks" on page 93
- "Configuring VPN networks on iOS devices" on page 94
- "Collecting audit events for Android devices" on page 126

Using labels for application groups

From the Core Admin portal **Device & Users > Labels** page, you can use labels for devices, apps, policies, and events. This process forms a group. For example, you might create a label called "Executives" to tag devices belonging to employees at the executive level. You can then locate all of these devices quickly in a search, or apply policies based on whether a device has this label.



While key tasks are listed here, see the section "Managing Labels" in *Getting Started with Core* for all tasks and information.

The following system labels are always available, by default:

TABLE 7. CORE DEFAULT LABELS

Label	Description
All-Smartphones	Automatically applied to all devices at registration
Android	Automatically applied to registered devices that have the Android platform selected during registration
Company-Owned	Automatically applied to registered devices that have the Company check box selected during registration

TABLE 7. CORE DEFAULT LABELS (CONT.)

Label	Description
Employee-Owned	Automatically applied to registered devices that have the Employee check box selected during registration
iOS	Automatically applied to registered devices that have the iOS platform selected during registration
macOS	Automatically applied to registered Apple devices that have macOS selected during registration
Signed-Out	Automatically applied to any multi-user iOS device that does not have a signedin user
tvOS	Automatically applied to registered devices that have the tvOS platform selected during registration
Windows	Automatically applied to Windows 10 devices
Windows Phone	Automatically applied to Windows Phone devices

 You cannot delete default labels.

Understanding label types

Labels fall into the following categories:

- Filter
- Manual

Filter labels (also called *dynamic* labels) use specific criteria to define a group of devices. Manual labels have no criteria associated with them; you select each device associated with a manual label.

When you initially create a label, it is stored as a filter label. If you use the Advanced Search feature to specify the criteria for a label, then it remains a filter label. Otherwise, if you select devices in an Admin Portal screen and apply the label to them, then the label becomes a manual label.

Creating a label

To create a new label:

Procedure

1. From the Admin Portal, go to **Device & Users > Labels**.
2. Click **Add Label**. The Add Label window opens.

FIGURE 1. ADD LABEL WINDOW

3. Refer to the guidelines in ["Add label window" below](#) to complete the fields.
4. Click **Save**. You can now apply this label to devices, policies, and configurations. See ["Applying a device to a label" on page 49](#).

Add label window

The following system labels are always available, by default:

TABLE 8. ADD LABEL WINDOW

Field	Description
Name	<p>Enter a unique name that clearly identifies the purpose of the label. The following characters are allowed when entering a label name. All other characters, including spaces, are prohibited.</p> <ul style="list-style-type: none"> • Letters (uppercase and lowercase)

TABLE 8. ADD LABEL WINDOW (CONT.)

Field	Description
	<ul style="list-style-type: none">• Numbers (0-9)• Dashes (-)• Underscores (_)• Periods (.)• At sign (@)• Dollar sign (\$)• Hash tag (#)• Extended ASCII/UTF-8
Description	Provide additional meaning and usage information.
Type	By default, the type is Filter . Change it to Manual if you want to manually associate devices with the label.
Criteria	If the type is Filter, use the query builder to create a search expression that defines the devices to apply the label to. Alternatively, manually enter a search expression. The matching devices are automatically displayed.

Editing a label

In **Device & Users > Labels**, you can edit:

- The name and description of any existing label.
- The type of a label (manual or filter).
- The criteria of a filter label.

If you change a label's type from manual to filter, you can use the query builder to define the filter. However, if you are changing a filter label's criteria, only manual editing is available to edit the criteria. The query builder is not available.

You can determine the string for the criteria by first navigating to **Devices & Users > Devices** and clicking **Advanced Search**. Use the user interface to create the criteria string and then copy it for pasting into the **Edit Label** dialog. You cannot edit the criteria of pre-defined labels such as **All-SmartPhones**, **Android**, **iOS**, **Company-Owned**, and so on.

Procedure

1. From the Admin Portal, go to **Device & Users > Labels**.
2. Select a label.
3. Click **Actions > Edit Label**.
4. Edit the name and/or description. The label name must be unique.
5. Click **Manual** or **Filter** to change the label type.



You can't change the label type when it has devices assigned to it.

6. For filter labels, edit the criteria.
 - If the type was already **filter**, manually edit the criteria.
 - If you changed the type to **filter**, either use the query builder or manually edit the filter.
7. Click **Save**.

Deleting a label

To delete a label:

Procedure

1. From the Admin Portal, go to **Device & Users > Labels**.
2. Select the label you want to delete.
3. Click **Delete**.



Default labels cannot be deleted.

Applying a device to a label

Applying a device to a label tags the device as part of the associated group. When you specify a label for an action, you perform the action on all devices having the label.

Procedure

1. From the Admin Portal, go to **Device & Users > Devices**.
2. Select the check box for the device.
3. Click **Apply To Label** from the **Actions** menu.
4. Select the label to apply from the **Apply To Label** dialog.

Only labels that have not already been associated with this device will be displayed. For example, iOS devices are automatically applied to the iOS label, Android devices to the Android label, and so on. Also, automatic labels that are not applicable to this device do not appear in the list. For example, the Windows label and Windows Phone label will not appear for a device from a different platform.

5. Click **Apply**.

Removing a device from a label

Removing a device from a label removes the following from the device:

- The tag that makes it a part of the associated group
- Policies applied to that label
- Apps applied to that label
- iBooks applied to that label (iOS only)

Procedure

1. From the Admin Portal, go to **Device & Users > Devices**.
2. Select the check box for the device or devices.
3. Click **Actions > Remove From Label**.
4. Select the label from the **Remove From Label** dialog.
5. Click **Remove**.

Configuring an ECDSA client identity certificate for mutual authentication

For devices, the client identity certificate that the device presents to Core must use ECDSA (Elliptical Curve Digital Signature Algorithm). For this purpose, Core behaves as the Local CA that issues the client identity certificate.

Generating a self-signed certificate

To configure Core to generate the self-signed certificate:

Procedure

1. Log into the Admin Portal.
2. Go to **Services > Local CA**.
3. Select **Add > Generate Self-Signed Cert**.
4. Change the following fields:
 - a. **Local CA Name:** Enter a recognizable name to identify the self-signed certificate. This name will appear in the list of local certificate authorities in **Services > Local CA**.
 - b. **Key Type:** Select **Elliptical Curve**.
 - c. **Key Length:** Specify the key length. The values are P-256, P-384 or P-521.
 - d. **CSR Signature Algorithm:** The values are SHA256, SHA384 (the default), and SHA512.
 - a. **Issuer Name:** Requires an X.509 name. For example, CN=www.yourcompany.com, DC=yourcompany, DC=com.



If you have a registered DNS name that you use to send SMTP mail, a best practice is to use the domain component convention and the DNS name for the certificate name.

5. Click **Generate**. The **Certificate Template** window displays.
6. Click **Save**.

Configuring a Local Certificate Authority as an Intermediate CA

You can configure Core as a local certificate authority (CA) to act as an Intermediate CA. Use this option when your company already has its own certificate authority. Using Core as an Intermediate CA gives your mobile device users the advantage of being able to authenticate to servers within your company intranet; not just the Core system.

After you get the certificate from your certificate vendor, you can add the certificates to Core to create the intermediate CA.

Procedure

1. In the Core Admin Portal, go to **Services > Local CA**.
2. Click **Add > Intermediate Enterprise CA**.
3. Click **Browse** and navigate to the combined file.
4. Click **Open**.
5. Enter a name you can recognize in the **Local CA Name** field.
6. Click **Upload Certificate**. Your local certificate authority is now available to use. The local CA will be listed in **Services > Local**.

Creating a certificate enrollment setting for the Local CA

After you have generated the self-signed certificate, create a local CA certificate enrollment setting for the self-signed certificate.

Procedure

1. Log into the Admin Portal.
2. Go to **Policies & Configs > Configurations**.
3. Click **Add New > Certificate Enrollment > Local**.
4. For **Name**, enter brief text that identifies this certificate enrollment setting.
5. For **Local CAs**, select the certificate you created in "[Generating a self-signed certificate](#)" on the [previous page](#).

6. For **Subject**, enter **CN= <value>**, in which *value* is selected from information in ["Supported custom attributes for mutual authentication certificates"](#) on page 23.
7. Click **OK**.
8. Click **Save**.

Specifying the certificate enrollment setting for the client identity certificate

After you have created the certificate enrollment setting for the local CA, configure Core so that it is used to provide the client identity certificates for mutual authentication.

Procedure

1. In the Admin Portal, go to **Settings > System Settings > Security > Certificate Authentication**.
2. For Certificate Enrollment Setting, select the certificate enrollment setting you created in ["Creating a certificate enrollment setting for the Local CA"](#) on the previous page.
3. Click **Save**.

Updating Core

You can upgrade Core software using the System Manager. Updates are confirmed using digital signatures during the download process. When a matching signature is confirmed, the status for the update changes to **Reboot to Install**.



Caution: Do not use the CLI method for upgrading.

To upgrade MobileIron Core software using the System Manager:

Procedure

1. Log into System Manager.
2. Go to **Maintenance > Software Updates** to display the **Software Updates** options.
3. Go to the **Software repository configuration** group.
4. Enter the credentials assigned by MobileIron Support.

5. Click **Apply > OK**.
6. Click **Check Updates** to show a list of the available updates.
7. Select the update you want.
8. Click **Download Now** if you want to download the update now and complete the installation at a later time.
9. Refresh the screen and click **Check Updates**. After the download is complete, the status for the update changes to **Downloaded**.
10. Click **Validate** to validate the database and select one of the following options:
 - **Validate Database structure (schema)** to verify that the existing database has the right database structure to proceed with upgrade.
 - **Validate the Database structure and Data** to copy the database to a temporary database to run the validation then click **Yes** to stop core services, (required for validation). Validating the database with data can take up to 4 hours, depending on the database size.

The Validation Status include the following options:

- **Not Running**
- **Validation Running**
- **Validation Failed**
- **Validation is Successful**

If the validation fails, do not proceed with the upgrade and contact MobileIron Support.



Validation is optional, but highly recommended. It alerts you to any problems that can happen during the upgrade process and can avoid the upgrade if the **Validate DB** returns errors. When the database validations has no errors, then you can proceed with upgrading the environment.

11. Refresh the screen and click **Check Updates**.
12. Click **Stage for Install** when you are ready to install.
 - If you have already downloaded the selected update, the system stages the update for installation.
 - If you did not previously download the selected update, it is downloaded and staged for installation.

13. After the software update has been staged for installation, the status for the update changes to **Reboot to Install**. You can now install the update by rebooting the system. If the status of an update is not **Reboot to Install**, rebooting the system will not install the update.
14. Select **Maintenance > Reboot** to reboot MobileIron Core.



To successfully install the update, you must reboot after the status is **Reboot to install**.

15. Continue with Verifying the upgrade is complete.

Verifying the upgrade is complete

To verify that the upgrade is complete:

Procedure

1. Go to the MobileIron Core System Manager: <https://<FQDN>:8443/mics>
2. Select **Maintenance > Software Updates**.
3. Confirm that the current version is correct.



Important: Under no circumstances should you interrupt the upgrade. Contact MobileIron Technical Support if you need assistance. Once this upgrade procedure is complete, it may take up to 5 minutes for MobileIron Client apps to display in the App Catalog page.

Provisioning AE devices in a closed network or AOSP deployment

Administrators register Android Enterprise (AE) devices by registering a “work profile” and by provisioning “work-managed” devices on a master device that is running the **Provisioner** app. Using this method, you can provision Android devices using a quick response (QR) code in a closed network or with a AOSP (Android Open Source Project) deployment.

Understanding AE device provisioning requirements

To provision an Android Enterprise device to be a work-managed device, the following settings and requirements must be met:

- The devices must be Android Enterprise-capable.
- You will need to download and install the **Provisioner** app on the master device. The Provisioner app is available from Google Play.
- The required Android Enterprise **configuration** is defined and applied to a recommended label.
- Android Enterprise is enabled on the server.
- In Devices & Users > Add Single Device, the **Include Registration PIN only for Android Company-Owned Device Enrollment** field is selected.
- In Settings > Device Registration, the **Managed Devices / Device Owner** field is set to one of these options:
 - **Password**
 - **Pin**
 - **Password and Pin**
- In **Settings > Device Registration**, the option **Display QR Code and Registration URL** is enabled. To disable sending users a QR code and registration URL, see ["Disabling the QR code and registration URL" on page 59](#).

Provisioning AE devices to become work-managed devices

This section applies to Work Managed Devices and Work Profile on Company Owned Devices.

Before you begin

Before beginning these tasks, see "[Setting up Core with a closed network / AOSP deployment](#)" on page 27.

Procedure

1. Using the Android master device, download the **Provisioner** app from [Google Play](#) and install the app.
2. Launch Provisioner on the master device.
3. Select **QR code** for the Provisioning method.
4. Tap **App for Provisioning**, and choose the client app to be installed on the provisioned device:

TABLE 9. SELECT A CLIENT APP

Select this client app:	To register with this EMM server:
Mobile@Work	Core

5. Fill out the remaining fields in the Provisioner app using these guidelines.:



Only the fields relevant to the selected provisioning method display.

TABLE 10. CONFIGURING THE PROVISIONER APP

Field	Value
Select app for provisioning	Mobile@Work
Time Zone	Enter the time zone to be configured on the device
Locale	Enter the locale to be configured on the device
Enable All System Apps	Click the check box to enable all system apps
Wi-Fi Network SSID	Enter the Wi-Fi SSID the target device is to use
Wi-Fi Security Type	Enter the Wi-Fi security type
Wi-Fi Password	Enter the password for the Wi-Fi

TABLE 10. CONFIGURING THE PROVISIONER APP (CONT.)

Field	Value
Bulk Enrollment	Bulk enrollment is optional along with the hostname and username. Optionally click the Quick Start check box to use Quick Start feature. If a username is entered or Quick Start is checked, then a hostname is required.

6. Tap **Continue**.
7. The screen **Scan this QR code!** appears on the master device.
8. Configure QR Code provisioning:
 - a. Confirm that the target device is displaying the **Android Welcome** screen.
 - b. **Tap** the Android Welcome screen on the target device **6 times** on the same place on the screen.
 - c. You will be prompted to **configure a WiFi network** so the setup wizard can download a QR code reader to the target device.
 - d. After the QR code reader is downloaded, the **camera** launches.
 - e. Hold the target device a few inches above the master device until it **scans the QR code successfully**. The setup wizard will then proceed to download the client app. If it is unable to download the client app, it will automatically do a factory reset.
 - f. You can continue to provision additional devices by scanning the QR code on the master device. The target device must have a camera ready to scan, and the master device must show the **Scan this QR code!** screen.

The QR code can also be exported by tapping the **Share** icon. The options offered for exporting will vary by device.

Creating a new Android Enterprise configuration

Administrators will need to create a new Android Enterprise configuration for the closed network or for an AOSP deployment, and then add the new configuration to a label.

Procedure

1. Go to **Policies & Configs > Configs**.
2. Click **Add New > Android Enterprise Setting**. The New Android Enterprise (all modes) Setting dialog box opens.
3. Enter a **Name** and **Description**.
4. Select the **Enable Closed Network/AOSP deployment** check box. All other options in the dialog box become hidden except **Enable Runtime Permissions**.
5. Click **Save**.
6. Apply a label to the configuration.



If two Android Enterprise configurations (one with Android for Work and one with AOSP) are pushed to the same device, the Android for Work configuration takes priority.



As Firebase Cloud Messaging (FCM) services will not be available, Ivanti recommends administrators to modify the **Sync Policy > Sync Interval** field to 30 minutes or to the lowest value as needed by the administrator.



Once the **Enable Closed Network/AOSP deployment** check box in the Android Enterprise Setting (configuration) dialog box is enabled, after saving, the option will be grayed out. There is no option for the administrator to disable it. As a workaround, the administrator should either remove the label or delete the configuration.

Disabling the QR code and registration URL

When new users are invited to register with Core, a QR code and registration URL display by default. If your organization prefers not to show users a QR code and registration URL, an administrator can disable the feature from the **Device Registration** page of the Core admin portal.

Procedure

1. Go to **Settings > System Settings > Users & Devices > Device Registration** page.
2. Deselect **Display QR Code and Registration URL** by clicking it.
3. Click **Save**.

Configuring Knox mobile enrollment

Core supports using the Samsung Knox Mobile Enrollment (KME) process to register qualified Samsung devices with Core. Using Samsung's Knox Mobile Enrollment process, once the process is set up, qualified devices are automatically enrolled and registered to Core when the end user activates the device for the first time.



Mobile@Work for Android is automatically installed during the enrollment process.

This section describes the following tasks:

- ["Prerequisites for Knox mobile enrollment" below](#)
- ["Creating the Knox mobile enrollment profile" on the next page](#)
- ["Assigning the KME profile to devices" on page 64](#)

Prerequisites for Knox mobile enrollment

Knox mobile enrollment requires the following items:

- Samsung devices running **Android 8** or newer versions, including a master device for the provisioning.
- A **CSV file** that provides a list of device IMEI numbers or serial numbers, and optionally, a username and a registration PIN and/or password. If the username and PIN or password is not included in the CSV file, the user must provide them.



If you configured registration to use a PIN, include a PIN in the registration file.

If you configured registration to use a password, include a password.

If you configured registration to use both a password and a PIN, include **only one of them** in the CSV file.

- A Samsung Knox account and use of the Samsung Knox Mobile Enrollment portal

You configure the registration requirements on the Admin Portal from:

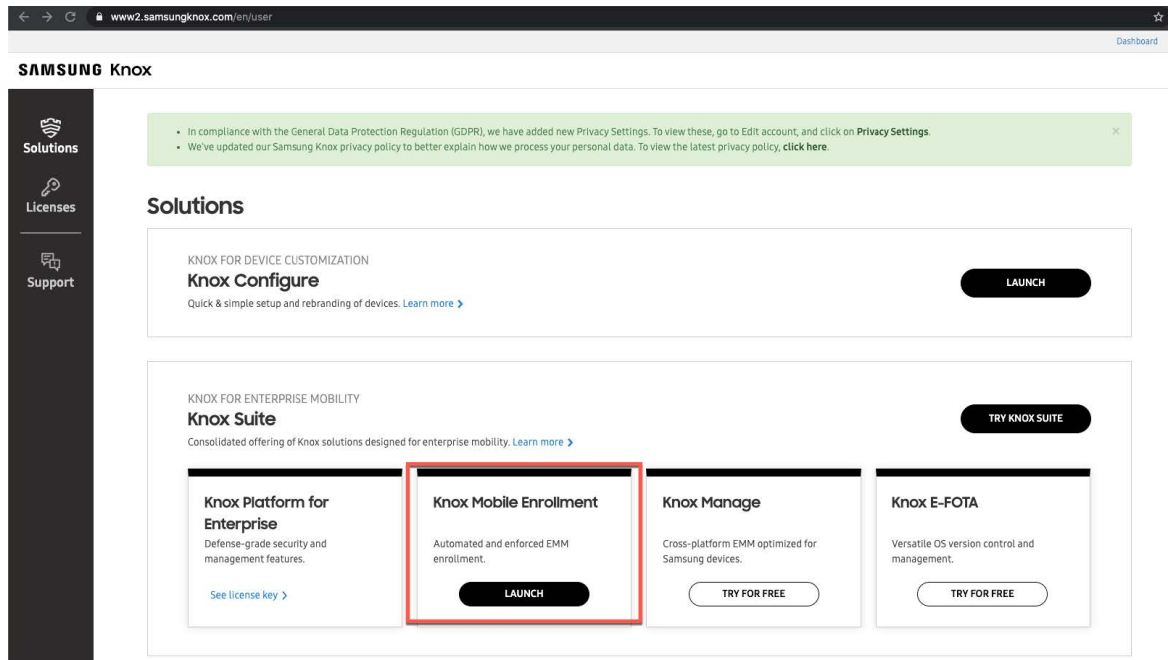
Settings > System Settings > User & Devices > Device Registration > Zero Touch and Samsung Knox Mobile Enrollment.

Creating the Knox mobile enrollment profile

Procedure

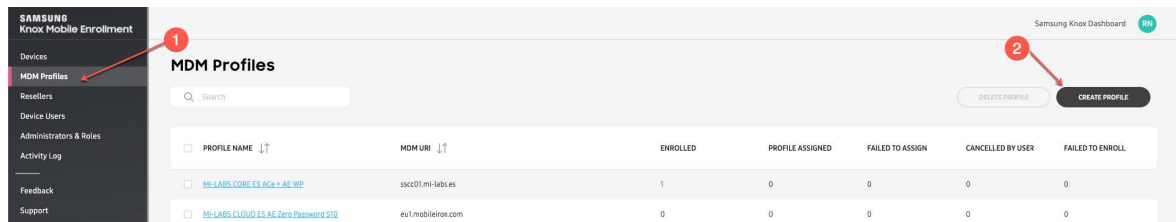
1. Log into <https://samsungknox.com> with an admin account, and within the **Knox Mobile Enrollment** block, click **Launch**. The Samsung Knox Mobile Enrollment dashboard displays.

FIGURE 1. SAMSUNG KNOX MOBILE ENROLLMENT DASHBOARD



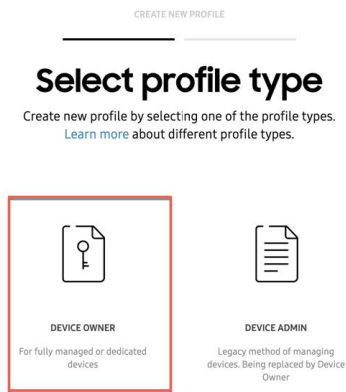
2. In the left pane, choose **MDM Profiles** (1). The MDM Profiles page displays.
3. On the far right of the page, click **Create Profile** (2). The Create new profile page displays.

FIGURE 2. CREATE NEW PROFILE PAGE



4. In the **Select profile type** section, select **Device Owner** (for fully-managed or dedicated devices). This will allow you to deploy Android Enterprise for work managed devices (also known as **Device Owner** mode) or fully-managed devices with work profile mode (also known as **COPE** mode).

FIGURE 3. SELECT PROFILE TYPE - DEVICE OWNER



5. In the **Device Owner profile details** section, enter the following information:
 - Under **BASIC INFORMATION**, name the profile and enter an optional description.
 - Under **MDM INFORMATION > Pick your MDM**, select **MobileIron**.
 - In the **MDM Agent APK** field, enter the Core Android application package (APK) URL:
<https://support.mobileiron.com/android-client-nfc/mi/mi-android-nfc-latest.apk>
6. In the **Device Owner profile settings** section, set your MDM configuration and device settings.

Under **DEVICE SETTINGS**:

- a. **System applications**: Decide if users will see pre-installed system apps or not:
 - **Disable system applications** - For **Device Owner** deployments, Ivanti recommends that you disable system apps, so users will see only those apps pushed by the administrator.
 - **Leave all system apps enabled** - For **COPE** deployments, Ivanti recommends that you leave all system apps enabled, so users can see all system apps on the personal side of the device.
- b. If you have an optional legal agreement you want your users to see, click **ADD LEGAL AGREEMENT** to attach it to the profile.

- c. In the **Company Name** field, enter your company name. This will display to users during enrollment.

Under **MDM CONFIGURATION**:

- a. Leave the **Custom JSON Data** field blank, unless you need to pass custom device attribute values to Core during the registration process. For more information, see Adding custom device attribute values
 - b. Do not enable **Dual DAR** for this profile.
 - c. If you plan to enroll devices using a QR code, click **ADD A QR CODE** to create one for the profile.
7. Click **Create** to save the profile.

Adding custom device attribute values

KME allows administrators to specify custom attributes that will be passed to Mobile@Work during enrollment. This is helpful if you have configurations on MobileIron UEM with filters based on the values of custom device attributes.

FIGURE 4. CUSTOM JSON DATA EXAMPLE

MDM CONFIGURATION

Custom JSON Data (as defined by MDM) ⓘ

```
{
  "token": "h5a64vn56",
  "quickStart": true
}
```

46 / 2000

Procedure

1. In the **Device Owner profile settings > MDM CONFIGURATION > Custom JSON Data** field, add your attributes in json format. For example:

```
{
  "key1": "COPE",
  "value1": "1",
  "quickStart": true
}
```

2. Validate the json structure before pasting it to the KME profile. Do this from this site: <https://jsonlint.com/>
3. Click **Create** to save the profile with the custom JSON data.

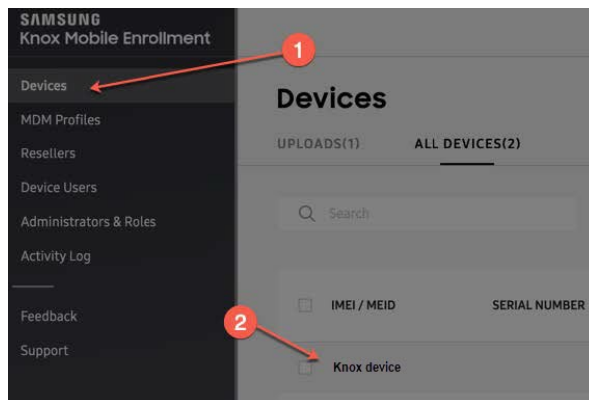
Assigning the KME profile to devices

When all the configuration tasks are complete, you need to assign the KME profile to compatible devices.

Procedure

1. From the [Samsung Knox Mobile Enrollment dashboard](#), click **Devices** from the left pane (1). The **Devices** page displays.

FIGURE 5. DEVICES PAGE



2. Select the target devices (2) and select your profile in the **MDM Profiles** dropdown menu (3).

FIGURE 6. DEVICE DETAILS

Device Details ✕

Device ID
Knox device 55

Model
SM-G973F

Status
● Enrolled

Submitted
18 Jul 2019 12:45:43
by Knox Deployment application

Edited
22 Jul 2019 14:00:22

MDM Profiles

MI-LABS CLOUD ES AE Zero Password S10

Profile changes will not take effect until the device is enrolled again

Tags

Bluetooth Raul's S10 ✕ Add a tag

User ID

Password

< Previous Next >

3. Optionally, add tags to allow identifying devices by additional criteria.

i Leave the **User ID** and **Password** fields blank. They are not relevant to this deployment.

4. Click **Save** (4).

Android-related configuration tasks

These tasks relate to configuring Core capabilities specific to Android devices.

Deploying Mobile@Work for Android	66
Configuring the Android warning banner	75
Configuring the lockdown policy for Android devices	76
Configuring allowed app sources for Android Samsung Knox devices	76
Wi-Fi settings for Android devices	79
Disabling the Developer options menu on Android devices	85
Prohibiting device users from unenrolling	86
Disabling biometric authentication on Android devices	86
Quarantining an Android device based on its OS version	87

Deploying Mobile@Work for Android

This section includes information and references to information about deploying Mobile@Work for Android.

Certified MDMPP Android client

Mobile@Work (specifically versions after 11.2.1.0) for Android is the certified MDMPP (Mobile Device Management Protection Profile) client for use with Core 11.

Supported devices

Mobile@Work (specifically versions after 11.2.1.0) for Android is supported with the devices listed in "[Device types in Common Criteria evaluation](#)" on page 4.

Differences between certified Mobile@Work for Android and Mobile@Work for Android in Google Play

The certified MDMPP Mobile@Work for Android differs from the Mobile@Work for Android in Google Play as follows:

Where the app is available

The certified MDMPP Mobile@Work 11 (specifically versions after 11.2.1.0) for Android is available at: http://support.mobileiron.com/fed_client.

Installation procedure

A device user installs the certified MDMPP Mobile@Work for Android as a side-loaded app, as described in ["Installing the certified MDMPP Mobile@Work for Android on a device" below](#).

Location service restriction behavior

The lockdown policy on Core has an option called **GPS**. When you select **Enable** for this option, all location services are disabled on the device, regardless of the source of the service (for example, GPS-based location services, network-based location services, or Wi-Fi-based location services). Furthermore, the lockdown policy option **GPS User Control** is ignored.

Registering to Core only with trusted SSL certificates

Certified Mobile@Work for Android allows registering to a Core only if Core uses a trusted SSL certificate for device registration. That is, the certificate chain is strictly validated. Entering the Core hostname into the whitelist has no impact.

Android Quick Setup policy usage

You cannot use the Android Quick Setup policy with the certified MDMPP Mobile@Work for Android. This policy is available only with the non-certified Mobile@Work for Android.

Verification of signed HTTPS responses from Core

When using mutual authentication, Core digitally signs HTTPS responses that it sends to Mobile@Work for Android. Core signs the responses using the Portal HTTPS certificate that you upload in the System Manager at **Security > Certificate Mgmt**. The Portal HTTPS certificate is used to automatically sign all payloads, such as policies and configurations, and commands, sent to the device. The MDMPP-certified Mobile@Work verifies the signature. If the verification fails, Mobile@Work ignores the response.

Installing the certified MDMPP Mobile@Work for Android on a device

Two methods are available for you to install the certified MDMPP Mobile@Work for Android on a device.

Install from a web site

Use this installation method if your device users will install Mobile@Work themselves.

The overall steps for installing from a web site are:

- You put the certified MDMPP Mobile@Work for Android APK file on a website and disclose its URL to your device users.
- Device users put the URL in a browser on their device to download the APK file. They tap on the downloaded APK file to install it.

Procedure

1. Download the Mobile@Work APK file from http://support.mobileiron.com/fed_client/11.2.1.0-30 to your computer.
2. Upload the Mobile@Work APK to a network location accessible to your device users.
3. Provide the URL to device users.

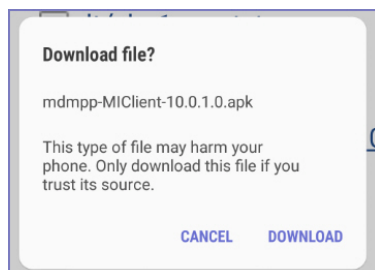
Each device user does the following:



The screen shots in these steps are examples to illustrate the procedure. Actual screen shots depend on the device.

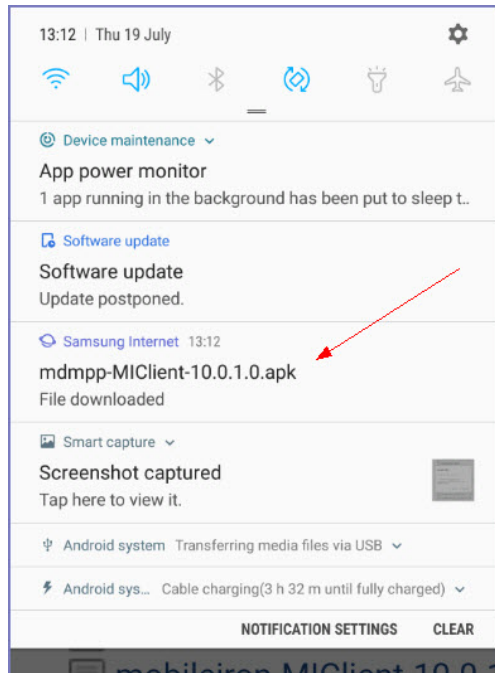
Procedure

1. Enter the URL that you provided into the address bar of a browser on the device.
2. When the browser has loaded the URL, it prompts the device user to accept the APK file, as in this example:

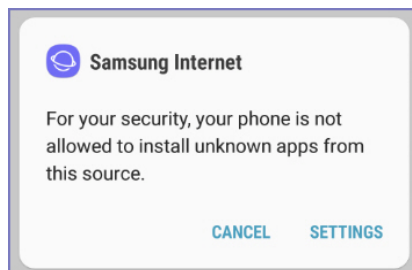


3. Tap **Download**.
4. The download of Mobile@Work begins in the background.

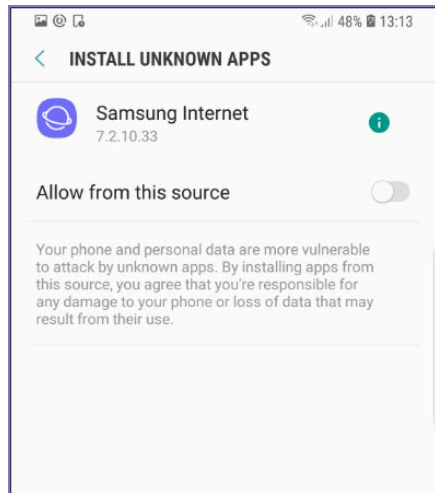
5. Pull down the device's notification bar to see the download progress, which is completed in this example.



6. When the download is complete, tap the notification to begin the installation.
7. If the **Unknown sources** setting is not enabled already, a prompt appears to go to the device's settings to enable it.

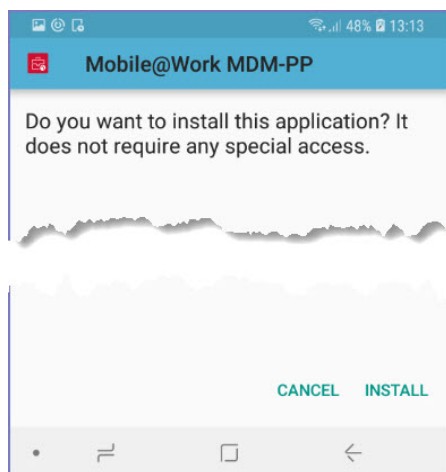


8. Tap **Settings**.



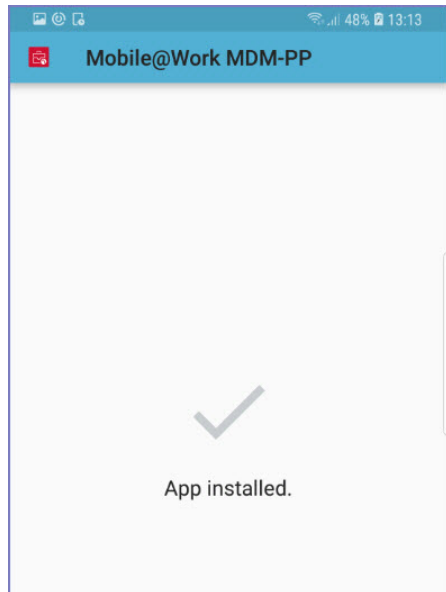
9. Tap **Allow from this source**.

10. A request is presented to install the app.



11. Tap **Install** and follow further installation instructions.

12. When the installation completes, the following screen displays.



Mobile@Work is now installed and ready to use on the device.



Ivanti recommends that the device user disables **Unknown sources** in the device's settings if it is still enabled. This setting prohibits app installation from unknown sources such as a web site or SD card. You can also apply a lockdown policy that selects **Disable** for the **Unknown sources** option.

Install with Android Debug Bridge

Use this installation method if you will be installing Mobile@Work on many devices yourself.

This method uses the Android Debug Bridge (ADB), which you install and run on your computer. The overall steps for each device are:

- Allow USB debugging on the device.
- Attach ADB on your computer to the device.
- Run an ADB command to install Mobile@Work on the device.

Before you begin the detailed steps for each device (listed below), complete the following prerequisites:

1. Download the Mobile@Work APK file from http://support.mobileiron.com/fed_client/10.0.1.0-45 to your computer.
2. Install Android Studio, which contains ADB, on your computer. For information on installing and using ADB, see:
 - <http://developer.android.com/sdk/installing/index.html?pkg=tools> to download the Android Stand-alone SDK tools.
 - <http://developer.android.com/tools/help/adb.html> for information about using ADB.



Attaching the computer to a device sometimes requires USB drivers on your computer. See <http://developer.android.com/tools/help/adb.html> for more information or work with the device manufacturer to get the appropriate drivers. Some manufacturers make the drivers available on their support web site.

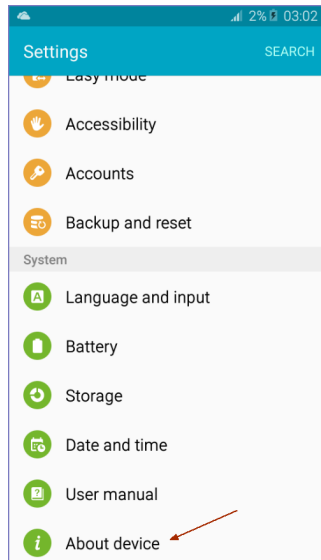
For each device, do the following:



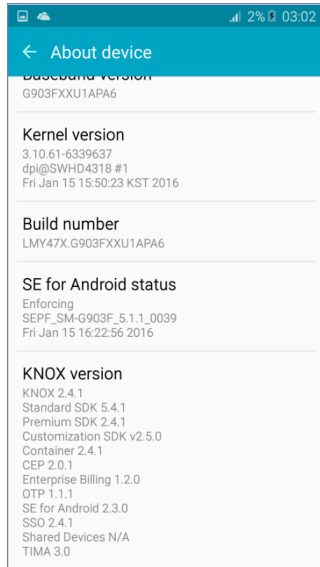
The screen shots in these steps are examples to illustrate the procedure. Actual screen shots depend on the device.

Procedure

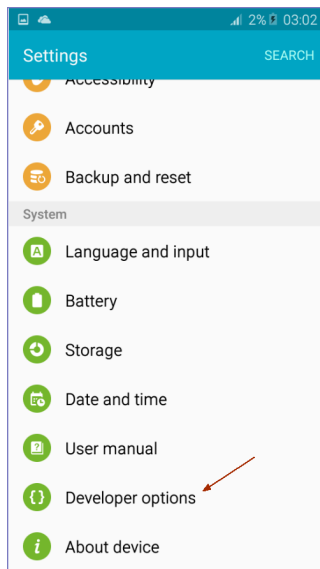
1. On the device, go to **Settings**.



2. Tap **About device**.

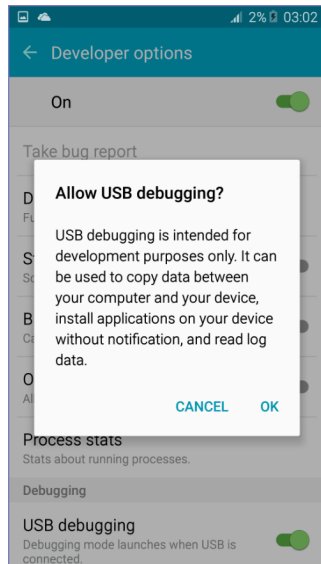


3. Tap **Build number** 5 to 10 times. This action causes the **Developer options** setting to be available.



4. Go to **Settings > Developer options**.

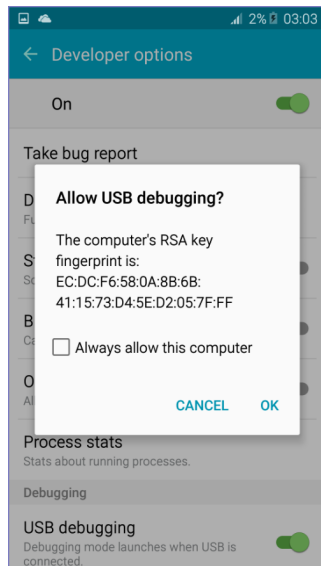
5. Tap **USB debugging**.



6. Tap **OK**.
7. Attach the computer running ADB to the device.



Attaching the computer to the device sometimes requires USB drivers on your computer. See <http://developer.android.com/tools/help/adb.html> for more information.



8. Tap **OK** to allow USB debugging with the computer.

9. On the computer, in the command prompt, enter the following command:

```
adb install <file path>
```

where `<file path>` is the path to the Mobile@Work APK on your computer.

For example, if the APK file is named **MIClient-latest.apk** and you have put it into the `/tmp` directory on your computer:

```
adb install /tmp/MIClient-latest.apk
```

ADB installs Mobile@Work. ADB outputs lines similar to the following in the command prompt:

```
8881 KB/s (12542951 bytes in 1.379s)
pkg: /data/local/tmp/MIClient-latest.apk
Success
```

Mobile@Work is now installed and ready to use on the device.

Configuring the Android warning banner

The Android warning banner appears before a device user is prompted to enter the device passcode for the first time after a device reboot.

Procedure

1. In the Admin Portal, go to **Policies & Configs > Policies**.
2. Select a privacy policy or add a new one.
3. Scroll down to **Android Warning Banner on the Device Reboot**.
4. Select **Enable Warning Banner**.
5. Enter the text you want to display in the warning.
6. Click **Save**.
7. If the policy is not already applied to a label for the target Android devices, apply a label (**Actions > Apply To Label**).

Configuring the lockdown policy for Android devices

The lockdown policy provides settings to restrict device access to specified features, including the following which can be disabled for Common Criteria mode (Bold text indicates the field on the lockdown policy):

- **Camera:** Camera access
- **NFC:** NFC access
- **Microphone** in **Android** section: Microphone access
- **Cellular Data** in **Samsung SAFE** section: Cellular data access

If you disable both cellular data and Wi-Fi on a device, MobileIron Core can no longer communicate with the device. The device may need a factory reset to restore functionality.

- **Bluetooth** and **Tethering - Bluetooth** in **Samsung SAFE** section: Remote access
- **Developer options** in **Samsung SAFE** section: Developer mode
- **USB Mass Storage:** USB mass storage mode
- **Tethering - USB** and **Tethering - Wi-Fi** in **Samsung SAFE** section: Unauthenticated hotspot and USB tethering
- **GPS:** Location services
- **Management Removal** in **Samsung SAFE** section: Device administration



Disable this setting to prohibit device users from unenrolling.

Configuring allowed app sources for Android Samsung Knox devices

The lockdown policy allows you to configure allowed app sources for Android Samsung Knox devices. Specifically:

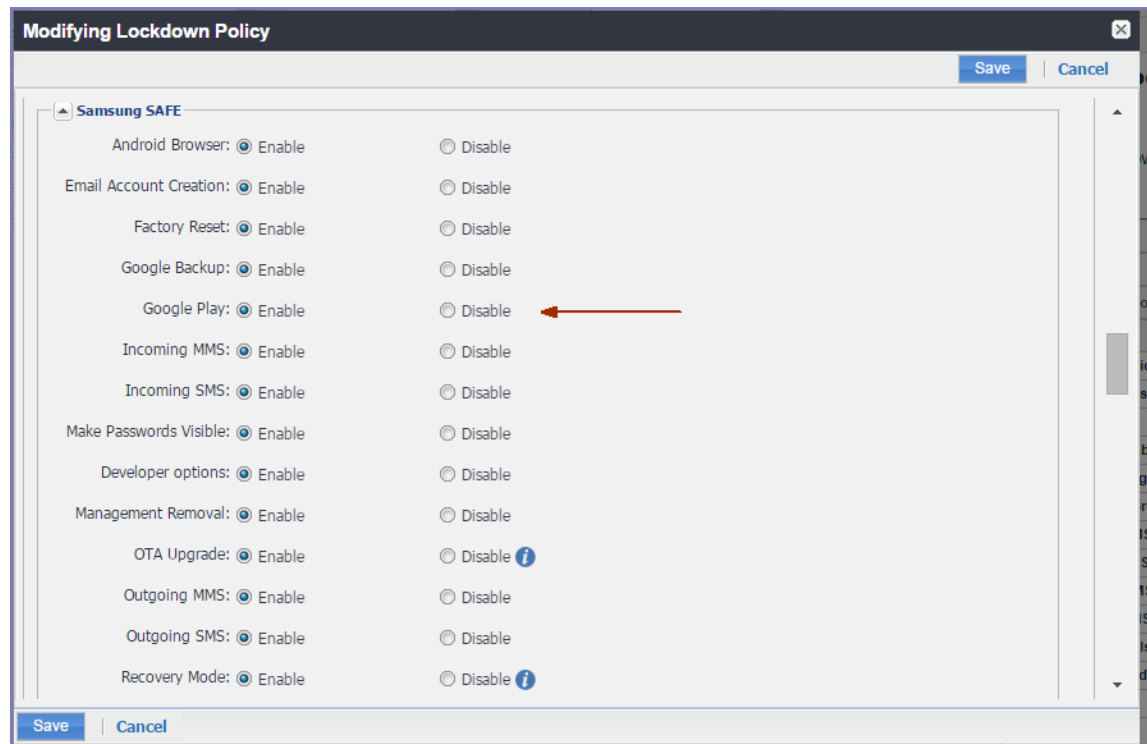
- You can disable Google Play by selecting the appropriate radio button.
- You can add the Samsung Apps store app to the list of restricted apps, which means that the app cannot be installed or run on the device. Without the Samsung Apps store app, the Samsung app store is not available as an app source.

If you choose to disable Google Play or the Samsung Apps store app, do the following:

Procedure

1. In the Admin Portal, go to **Policies & Configs > Policies**.
2. Select a lockdown policy or add a new one.
3. Scroll down to the **Samsung SAFE** section.

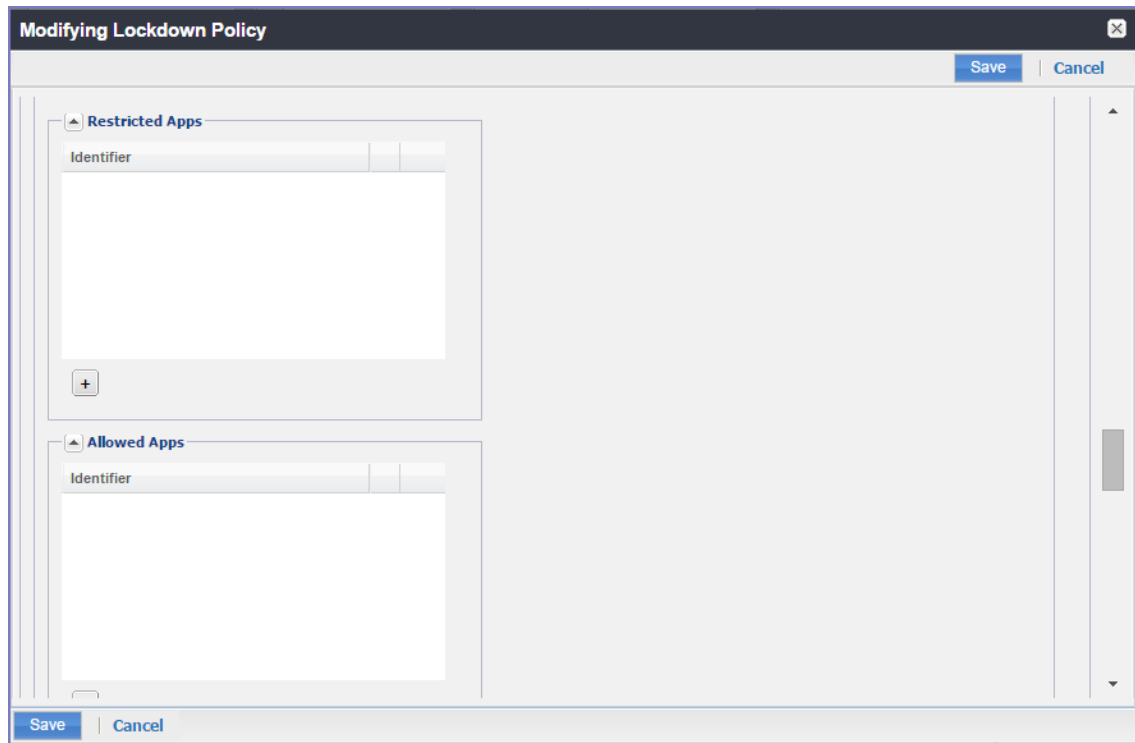
FIGURE 1. LOCKDOWN POLICY SAMSUNG SAFE SECTION



4. For **Google Play**, select **Disable**.

5. Scroll further down in the **Samsung SAFE** section to the **Restricted Apps** section.

FIGURE 2. RESTRICTED APPS SECTION OF SAMSUNG SAFE SECTION



6. In **Restricted Apps**, click + .
7. In the **Identifier** field, enter **com.sec.android.app.samsungapps**.
8. Click **Save**.
9. Select the policy you just created.
10. Select **Actions > Apply To Label**.
11. Select the label that identifies the devices you want to target.
12. Click **Apply > OK**.

For more information about the lockdown policy, see "[Configuring the lockdown policy for Android devices](#)" on page 76.

Wi-Fi settings for Android devices

To configure wireless network access for Android devices:

Procedure

1. In the Admin Console, go to **Policies & Configs > Configurations**. Click **Add New > Wi-Fi** to create a new configuration.



Do not assign multiple Wi-Fi profiles to a device if the Network Name SSID (Service Set Identifier) differs only by case. For example, if one profile has an SSID value of "yourco" and another has an SSID of "YourCo," those two must not be assigned to the same device. Doing so will cause check-in problems, and full device details will not be properly recorded.

Android 10 devices

On Android 10 devices or supported newer versions, upon installation or upgrade, device users can configure Wi-Fi and location settings in specific modes.



Administrators are required to leave in all modes of deployment to enable Wi-Fi and MTD configurations to be successfully applied. This means having the Allow the user to turn on location sharing lockdown field selected (checked.)

The table below depicts the behavior changes in different configuration modes:

TABLE 11. WI-FI CHANGES IN SPECIFIC CONFIGURATION MODES

Item	Description
All modes	Disconnect Wi-Fi local action is disabled in all modes on Android 10 devices. For all modes of deployment, to enable Wi-Fi and MTD configurations to be successfully applied, the Allow the user to turn on location sharing lockdown field must be selected.
(Android Enterprise) <ul style="list-style-type: none">• Work Profile mode• Work Profile on Company Owned devices (Android 11 or supported newer versions)	Device users are requested to activate location for the device and for the managed profile. In order for administrators to update Wi-Fi and to have Mobile Threat Defense detect Wi-Fi-based threats, device users must activate location. If the device user chooses No, the device will be flagged with an unblocking error for non-compliance and Core will report a configuration error. Administrators will not be able to disable Wi-Fi through UEM configurations in Work managed device mode on Android 10 devices.

TABLE 11. WI-FI CHANGES IN SPECIFIC CONFIGURATION MODES (CONT.)

Item	Description
(Android Enterprise) <ul style="list-style-type: none"> • Work Managed Device (COPE) mode 	In the background, Core will turn on the location services setting without device user intervention. Wi-Fi and MTD configurations should be successful with no errors. If there is no MTD configuration or a Wi-Fi configuration, the device user can switch location service on or off.
Device Administrator (DA) mode	Wi-Fi configurations will not be supported and will show as Sent on the server with config error. MTD configurations will be still accepted for non-network threats but the Wi-Fi related threats will not work for Device Administrators and MAM. Administrators will not be able to disable Wi-Fi through UEM configurations in Device Administrator mode on Android 10 devices.
Kiosk mode	Administrators wanting users to enable/disable Wi-Fi but not connect to any other Wi-Fi network settings are not supported. Options available to administrators are: <ul style="list-style-type: none"> • Scenario 1 - Administrators wanting users to enable/disable Wi-Fi and connect to any available Wi-Fi will need to have the following settings in Kiosk mode: <p>Lockdown settings:</p> <ul style="list-style-type: none"> ◦ Allow Wi-Fi (de-selected) and Allow Wi-Fi to be configured (de-selected). <p>Kiosk Mode Settings:</p> <ul style="list-style-type: none"> ◦ Allow users to Access Wi-Fi Settings (selected). • Scenario 2 - Administrators wanting to block users from any Wi-Fi controls will need to have the following setting: <p>Lockdown settings:</p> <ul style="list-style-type: none"> ◦ Allow Wi-Fi (selected) ◦ Allow Wi-Fi to be configured (selected)

Using Wi-Fi priority values

Administrators can use the Wi-Fi priority feature to influence the network connections that devices make. The table below describes how to set Wi-Fi priorities to achieve the noted results.

TABLE 12. WI-FI PRIORITY VALUES

Desired outcome	Settings required
<p>Allow any Wi-Fi connection, equally</p>	<p>Default state.</p> <ul style="list-style-type: none"> • For Always Connect Device to Managed Wi-Fi in Lockdown policy: Disable • For Wi-Fi configuration: The Priority field is ignored
<p>Ensure that managed Wi-Fi networks always get priority over non-managed networks;</p> <p>Actively disconnect from unmanaged networks when a managed network is available.</p> <p>Do not allow users to override the Wi-Fi connection choice with an unmanaged network.</p>	<ul style="list-style-type: none"> • For Always Connect Device to Managed Wi-Fi in Lockdown policy: Enable • For Wi-Fi configuration: Set the Priority field for each Wi-Fi configuration. 1 = lowest, 100 = highest.

Wi-Fi authentication types

Core supports the following Wi-Fi authentication types:

- **Open authentication** – Open authentication allows any device to authenticate and then attempt to communicate with the access point. Using open authentication, any wireless device can authenticate with the access point, but the device can communicate only if its Wired Equivalent Privacy (WEP) keys match the access point's WEP keys.
- **Shared authentication** – Shared Key Authentication (SKA) is a process by which a computer can gain access to a wireless network that uses the Wired Equivalent Privacy (WEP) protocol. With SKA, a computer equipped with a wireless modem can fully access any WEP network and exchange encrypted or unencrypted data.
- **WPA Enterprise authentication** – Wi-Fi Protected Access-Enterprise (WPA-Enterprise) is a wireless security mechanism designed for small to large enterprise wireless networks. It is an enhancement to the WPA security protocol with advanced authentication and encryption.
- **WPA2 / WPA3 Enterprise authentication** – WPA-Enterprise uses TKIP with RC4 encryption, while WPA2-Enterprise adds AES encryption. WPA3 uses Simultaneous Authentication of Equals (SAE) to provide stronger defenses against password guessing. SAE is a secure key establishment protocol. WPA3-Enterprise provides additional protections for networks transmitting sensitive data by offering the equivalent of 192-bit cryptographic strength.

- **WPA Personal authentication** – WPA-Personal. Also referred to as WPA-PSK (pre-shared key) mode, this is designed for home and small office networks and doesn't require an authentication server. Each wireless network device encrypts the network traffic by deriving its 128-bit encryption key from a 256-bit shared key.
- **WPA2 / WPA3 Personal authentication** – WPA2 is currently the most secure standard utilizing AES (Advanced Encryption Standard) and a pre-shared key for authentication. WPA2 is backwards compatible with TKIP to allow interoperability with legacy devices. WPA3 Personal is available as a setting in the local browser user interface (UI). This personal authentication option is a more secure option than WPA2.

Supported variables for Wi-Fi authentication

You can use the following variables in fields that support variables.

- \$PASSWORD\$ (only supported in the password field)
- \$EMAIL\$
- \$USERID\$
- \$DEVICE_MAC\$
- \$NULL\$
- \$USER_CUSTOM1\$... \$USER_CUSTOM4\$ (custom fields defined for LDAP)

Custom attribute variable substitutions are supported.

Restricting Wi-Fi access to specific networks on Android devices

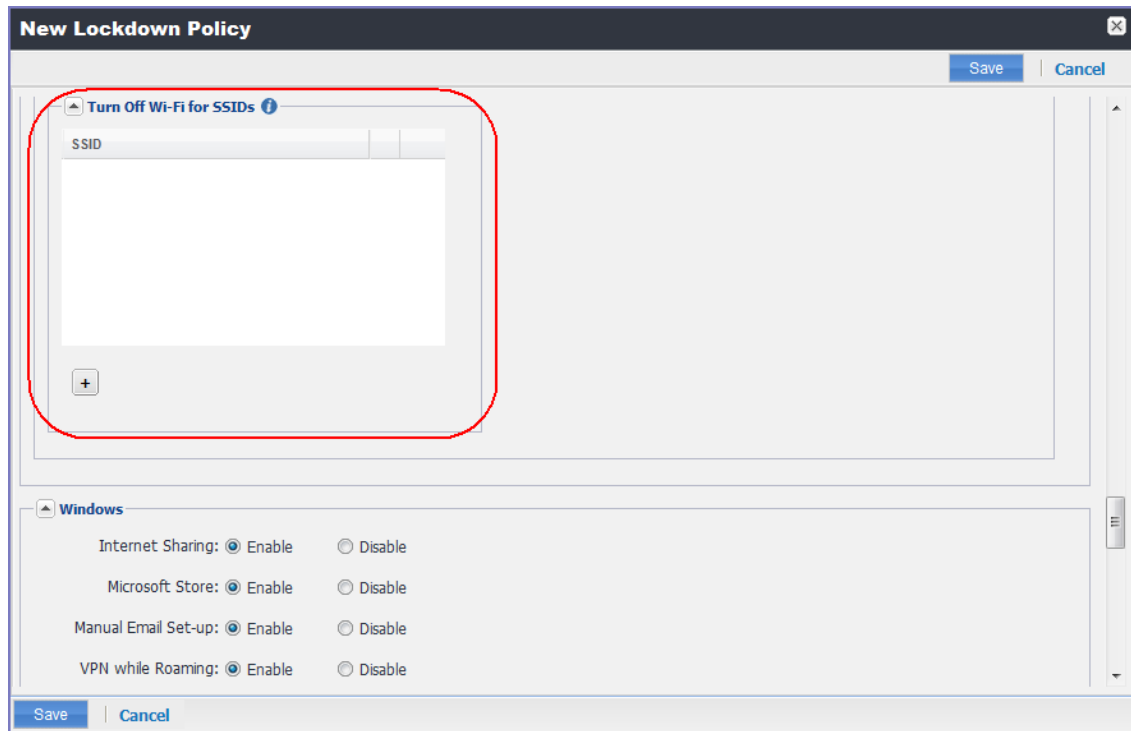
To restrict managed Android devices to specific networks, you first need to block all networks in the lockdown policy. Then you configure Core to allow specific Wi-Fi networks.

Procedure

1. In the Admin Portal, go to **Policies & Configs > Policies**.
2. Select **Add New > Lockdown**.

3. Scroll down to **Turn Off Wi-Fi for SSIDs**.

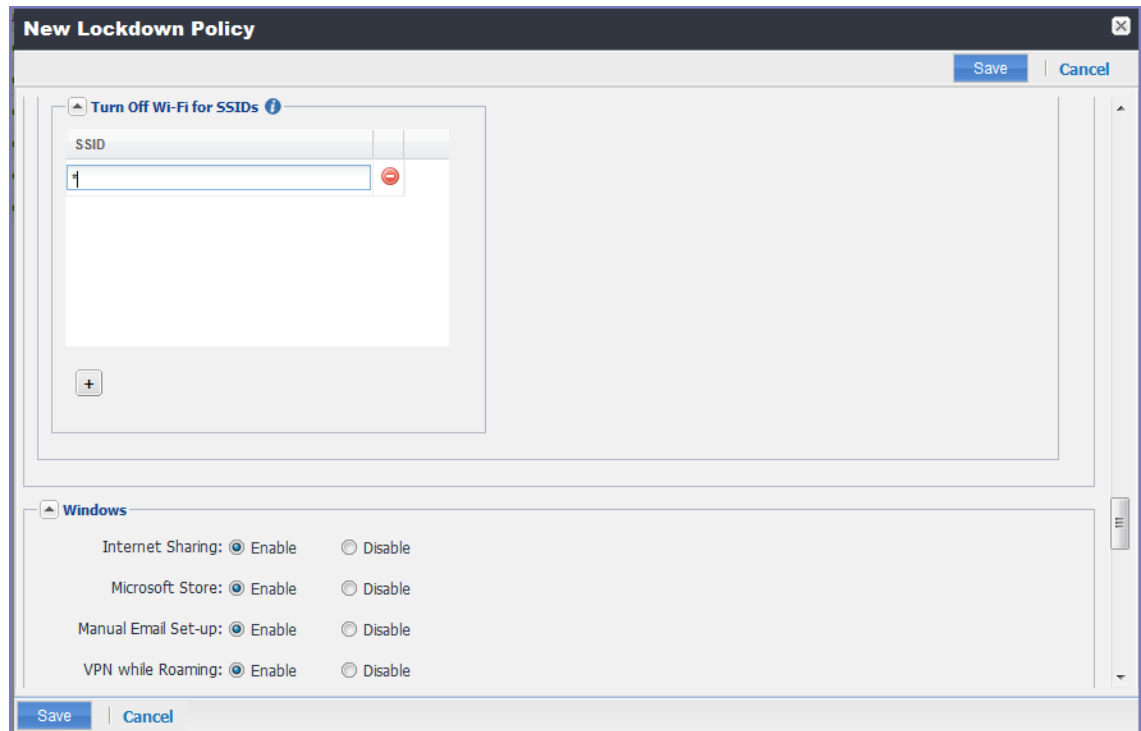
FIGURE 1. TURN OFF WI-FI FOR SSID CONFIGURATION WINDOW



4. Click the + button.

5. Enter * to indicate all SSIDs.

FIGURE 2. ASTERISK SIGNIFIES ALL SSIDS



6. Click **Save**.
7. Select the policy you just created.
8. Select **Actions > Apply To Label**.
9. Select the label that identifies the devices you want to target.
10. Click **Apply > OK**.
11. Go to **Policies & Configs > Configurations**.
12. Click **Add New > Wi-Fi**.
13. Create a Wi-Fi configuration for a network you want to allow.
14. Select the configuration you just created.
15. Select **Actions > Apply To Label**.

16. Select the label that identifies the devices you want to target.
17. Click **Apply > OK**.

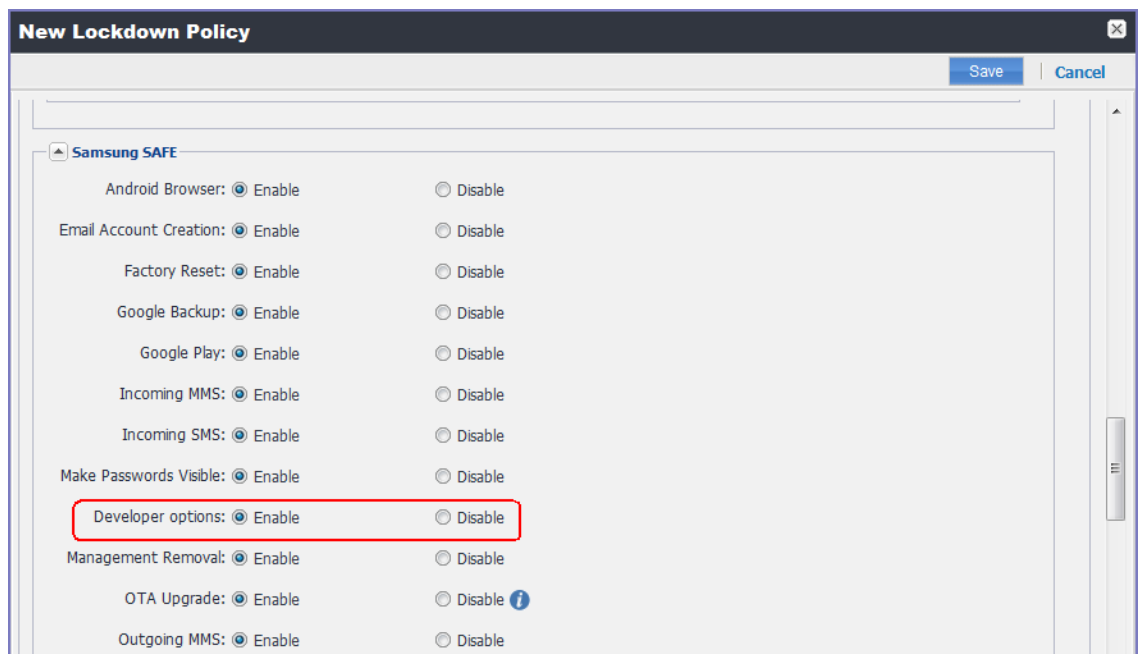
Disabling the Developer options menu on Android devices

To prevent device users from accessing the Developer options menu on Samsung Knox devices:

Procedure

1. In the Admin Portal, go to **Policies & Configs > Policies**.
2. Select **Add New > Lockdown**.
3. Scroll down to the **Samsung SAFE** section.

FIGURE 1. SAMSUNG SAFE DEVELOPER OPTIONS



4. Set **Developer options** to **Disable**.
5. Click **Save**.
6. Select the policy you just created.
7. Select **Actions > Apply To Label**.

8. Select the label that identifies the devices you want to target.
9. Click **Apply** > **OK**.

Prohibiting device users from unenrolling

You can prohibit device users from unenrolling their devices from Core on the lockdown policy for the appropriate devices.

Procedure

1. In the Admin Portal, go to **Policies & Configs** > **Policies**.
2. Select **Add New** > **Lockdown**.
3. Scroll down to the **Samsung SAFE** section.
4. Set the **Management Removal** field to **Disable**.

Disabling biometric authentication on Android devices

You can disable biometric authentication, including fingerprint and iris and face scanning, on Android devices.

Procedure

1. In the Admin Portal, go to **Policies & Configs** > **Policies**.
2. Select the appropriate security policy.
3. Click **Edit**.
4. For **Password Type**, select **Simple** or **Alphanumeric**. Either selection blocks fingerprint and iris and face scanning.
5. If you selected **Don't Care** for **Password Type**, no biometric authentication is blocked. However, you can block fingerprint, but not iris and face scanning using another field in the security policy.

To block fingerprint, in the **Android** section, select **Block Fingerprint (from Android 5.0 or Samsung MDM 5.3)**

6. Click **Save**.

Quarantining an Android device based on its OS version

To quarantine an Android device because its Android OS version is not updated, complete the following tasks:

- [Configure an Android OS version alert](#)
- [Define a custom compliance action](#)
- [Set up the security policy to trigger the compliance action when the violation occurs](#)

Configuring an Android OS version alert

To trigger an alert when a device is using a disallowed Android operating system:

Procedure

1. In the Admin Portal, go to **Logs > Event Settings**.
2. Select **Add New > Policy Violations Event**.
3. Enter a name for the event.
4. In the **Security Policy Triggers** section, under the **Android** heading, confirm that the app control alert **Disallowed Android OS version** found is selected.
5. Deselect all the other checkboxes.
6. In the **Apply to Labels** section, select the appropriate labels in the **Available** column, and click the right arrow to move them to the **selected** column.
7. Click **Save**.



When creating a policy violations event, you can choose to alert an administrator with an email. For Core to send this email, configure the email settings in the System Manager at **Settings > Email Settings**.

Defining a custom compliance action

To define custom compliance actions for your devices:

Procedure

1. In the Admin Portal, go to **Policies & Configs > Compliance Actions**.
2. Click **Add+** to open the **Add Compliance Action** dialog.
3. Enter a name for the compliance action.
4. Select **Enforce Compliance Actions Locally on Devices**.
5. In the **Alert** section, select **Send a compliance notification or alert to the user**.
6. In the **Block Access** section, select **Block email access and AppConnect apps**.
7. In the **Quarantine** section, select **Quarantine the device**.
8. Select **Remove All Configurations**.
9. Click **Save**.

Setting up the security policy to trigger the compliance action when the violation occurs

To create a security policy to trigger compliance when a violation occurs:

Procedure

1. In Admin Portal, go to **Policies & Configs > Policies**.
2. Select the security policy you want to work with.
3. Click **Edit**.
4. Scroll down to the **Access Control** section of the **Modifying Security Policy** dialog.
5. Under **For Android devices**, select the checkbox for **when Android version is less than**.
6. On the same line, in the dropdown list, select the custom compliance action that you just created.
7. On the same line, in the dropdown list for Android OS versions, select the appropriate OS version.

8. Click **Save**.
9. Click **OK**.
10. Confirm that the security policy is applied to a label that is also applied to the target devices.
 - a. Select the security policy.
 - b. Click **Actions > Apply To Label**.
 - c. If necessary, select the appropriate labels and click **Apply**.

iOS-related configuration tasks

These tasks relate to configuring Core for capabilities and that are specific to iOS devices.

Configuring the password policy for iOS devices	90
Configuring iOS restrictions	90
Wi-Fi settings for iOS devices	91
Configuring VPN networks on iOS devices	94
Updating the operating system for iOS devices	94
Setting up the Apps@Work web clip	95

Configuring the password policy for iOS devices

You can configure the iOS device password policy in the security policy on Core. When a password is mandatory, the device user must create a device password according to your requirements. The password policy allows you to specify the following password characteristics (Bold text indicates the field on the security policy):

- **Minimum Password Length:** Minimum password length
- **Minimum Number of Complex Characters:** Minimum password complexity
- **Maximum Password Age:** Maximum password lifetime
- **Password field set to Mandatory:** Screen lock enabled
- **Maximum Inactivity Timeout:** Screen lock timeout
- **Grace Period for Device Lock:** Number of authentication failures

Configuring iOS restrictions

You can specify lockdown capabilities for iOS devices in the **Restrictions** section. To configure iOS restrictions:

Procedure

1. Go to **Policies & Configs > Configurations > Add New > iOS and macOS > iOS Only > Restrictions.**

2. Select from the following restrictions:
 - Denying application installation (**Allow installing apps using Apple Configurator and iTunes**)
 - Enable/disable camera (**Allow use of camera**)
 - Enable/disable policy for display notification in the locked state (Show Notification Center in Lock screen)
 - Validate digital signatures in in-house apps (**Allow trusting new enterprise app authors**)
 - Revoke biometric template (**Allow Face ID/Touch ID to unlock device**)
3. Save your changes.

Wi-Fi settings for iOS devices

To configure wireless network access for iOS devices:

Procedure

1. In the Admin Console, go to **Policies & Configs > Configurations**. Click **Add New > Wi-Fi** to create a new configuration.



Do not assign multiple Wi-Fi profiles to a device if the Network Name SSID (Service Set Identifier) differs only by case. For example, if one profile has an SSID value of "yourco" and another has an SSID of "YourCo," those two must not be assigned to the same device. Doing so will cause check-in problems, and full device details will not be properly recorded.

Wi-Fi profiles and password caching

To make deployments easier, Core offers the option of caching a user's Wi-Fi password. This option is turned off by default. Cached passwords are encrypted, stored on Core, and used only for authentication. Note that the password must match the LDAP password in order for this feature to be of use.

Using Wi-Fi priority values

Administrators can use the Wi-Fi priority feature to influence the network connections that devices make. The table below describes how to set Wi-Fi priorities to achieve the noted results.

TABLE 13. WI-FI PRIORITY VALUES

Desired outcome	Settings required
Allow any Wi-Fi connection, equally	Default state. <ul style="list-style-type: none"> • For Always Connect Device to Managed Wi-Fi in Lockdown policy: Disable • For Wi-Fi configuration: The Priority field is ignored
Ensure that managed Wi-Fi networks always get priority over non-managed networks; Actively disconnect from unmanaged networks when a managed network is available. Do not allow users to override the Wi-Fi connection choice with an unmanaged network.	<ul style="list-style-type: none"> • For Always Connect Device to Managed Wi-Fi in Lockdown policy: Enable • For Wi-Fi configuration: Set the Priority field for each Wi-Fi configuration. 1 = lowest, 100 = highest.

Wi-Fi authentication types

Core supports the following Wi-Fi authentication types:

- **Open authentication** – Open authentication allows any device to authenticate and then attempt to communicate with the access point. Using open authentication, any wireless device can authenticate with the access point, but the device can communicate only if its Wired Equivalent Privacy (WEP) keys match the access point's WEP keys.
- **Shared authentication** – Shared Key Authentication (SKA) is a process by which a computer can gain access to a wireless network that uses the Wired Equivalent Privacy (WEP) protocol. With SKA, a computer equipped with a wireless modem can fully access any WEP network and exchange encrypted or unencrypted data.
- **WPA Enterprise authentication** – Wi-Fi Protected Access-Enterprise (WPA-Enterprise) is a wireless security mechanism designed for small to large enterprise wireless networks. It is an enhancement to the WPA security protocol with advanced authentication and encryption.
- **WPA2 / WPA3 Enterprise authentication** – WPA-Enterprise uses TKIP with RC4 encryption, while WPA2-Enterprise adds AES encryption. WPA3 uses Simultaneous Authentication of Equals (SAE) to provide stronger defenses against password guessing. SAE is a secure key establishment protocol. WPA3-Enterprise provides additional protections for networks transmitting sensitive data by offering the equivalent of 192-bit cryptographic strength.

- **WPA Personal authentication** – WPA-Personal. Also referred to as WPA-PSK (pre-shared key) mode, this is designed for home and small office networks and doesn't require an authentication server. Each wireless network device encrypts the network traffic by deriving its 128-bit encryption key from a 256-bit shared key.
- **WPA2 / WPA3 Personal authentication** – WPA2 is currently the most secure standard utilizing AES (Advanced Encryption Standard) and a pre-shared key for authentication. WPA2 is backwards compatible with TKIP to allow interoperability with legacy devices. WPA3 Personal is available as a setting in the local browser user interface (UI). This personal authentication option is a more secure option than WPA2.

Supported variables for Wi-Fi authentication

You can use the following variables in fields that support variables.

- \$PASSWORD\$ (only supported in the password field)
- \$EMAIL\$
- \$USERID\$
- \$DEVICE_MAC\$
- \$NULL\$
- \$USER_CUSTOM1\$... \$USER_CUSTOM4\$ (custom fields defined for LDAP)

Custom attribute variable substitutions are supported.

Restricting Wi-Fi access to specific networks on iOS devices

You can limit the Wi-Fi networks that iOS devices can join by specifying an option on the Core security policy. When you select the option, the networks are limited to those that are installed using a Wi-Fi setting on Core. A Wi-Fi setting results in Core pushing an MDM profile with the Wi-Fi configuration to the device.

Whitelisting Wi-Fi networks

You can limit the Wi-Fi networks iOS devices can join only to those Wi-Fi networks installed by profiles. This option enhances security, in that the Wi-Fi networks installed by profiles on iOS devices are secure, trusted networks. You enable this option in the security policy pushed to iOS devices.

After you push the Wi-Fi whitelist restriction to devices, the restriction will work only if Core has pushed at least one Wi-Fi network configuration to iOS devices using an MDM profile. The MDM profile with Wi-Fi configuration can only be pushed to devices through Core (and not through any other means, such as Apple Configurator). If Core removes all Wi-Fi configurations from the device, then the Wi-Fi whitelist restriction is removed as well.



Enabling and disabling this feature causes Core to push Wi-Fi configurations to devices. As such, try to minimize enabling and disabling this feature.

This feature applies only to supervised devices running iOS 10.3 or supported newer versions.

Procedure

1. Go to **Policies & Configs > Policies**.
2. Select the security policy for which you want to enable this feature.
3. Click **Edit**.
4. Select **Only join Wi-Fi networks installed by profiles (iOS 10.3 and later with supervised devices only)**.
5. Click **Save**.

Configuring VPN networks on iOS devices

In the Admin Portal, go to **Policies & Configs > Configurations** and click **Add New > VPN** to configure VPN access. Refer to "Managing VPN Settings" chapter of the *Core Device Management Guide for iOS and macOS Devices* for configuration information for the exact VPN solution you are using.

Updating the operating system for iOS devices

Before you begin

Be sure you have configured your iOS software update policy.

Procedure

1. Go to **Devices & Users > Devices**.
2. Select the devices whose operating system you wish to update.

3. Select **Actions > iOS and macOS > Update OS Software**. A confirmation dialog box opens.
4. Click **Confirm**.

Core sends the OS update command to devices. Core shows the status of the command in the **Update OS Software** window.

5. Click **OK**.

Setting up the Apps@Work web clip

Users on iOS devices use the Apps@Work web clip to install apps that are in the App Catalog of Core. To set up the Apps@Work web clip on iOS devices, see "Setting up Apps@Work for iOS and macOS" in the *Core Apps@Work Guide*.



An analogous procedure is not necessary for Android devices. On Android devices, Apps@Work is part of the Mobile@Work app.

Managing devices

These tasks relate to actions taken toward devices. In this chapter:

- ["Registering/Enrolling a device" on the next page](#)
- ["Locking a device" on page 98](#)
- ["Wiping a device" on page 98](#)
- ["Unenrolling a device" on page 98](#)
- ["The App Catalog" on page 99](#)
- ["Using labels for application groups" on page 45](#)
- ["Configuring mutual authentication" on page 21](#)
- ["Exporting device status events" on page 123](#)
- ["Collecting audit events for Android devices" on page 126](#)

Device tasks in other sections

Common Criteria

- ["Device types in Common Criteria evaluation" on page 4](#)
- ["Enabling Common Criteria mode for Samsung Knox devices" on page 12](#)

General Core configuration tasks

- ["Configuring the sync interval for a device" on page 42](#)
- ["Limiting the number of devices that users can enroll" on page 43](#)
- ["Querying hardware and software information about a device " on page 44](#)
- ["Querying the installed apps on a device " on page 44](#)
- ["Installing policies on a device" on page 44](#)

iOS-related configuration tasks

- ["Configuring the password policy for iOS devices" on page 90](#)
- ["Wi-Fi settings for iOS devices" on page 91](#)
- ["Configuring VPN networks on iOS devices" on page 94](#)
- ["Updating the operating system for iOS devices" on page 94](#)

Setting up Core with a closed network / AOSP deployment

- ["Provisioning the Android device" on page 31](#)
- ["Managing the closed network / AOSP devices" on page 32](#)

Provisioning AE devices in a closed network or AOSP deployment

- ["Provisioning AE devices in a closed network or AOSP deployment" on page 56](#)

Configuring Knox mobile enrollment

- ["Assigning the KME profile to devices" on page 64](#)

Android-related configuration tasks

- ["Configuring the lockdown policy for Android devices" on page 76](#)["Configuring allowed app sources for Android Samsung Knox devices" on page 76](#)
- ["Configuring allowed app sources for Android Samsung Knox devices" on page 76](#)
- ["Wi-Fi settings for Android devices" on page 79](#)
- ["Disabling the Developer options menu on Android devices" on page 85](#)
- ["Prohibiting device users from unenrolling" on page 86](#)
- ["Disabling biometric authentication on Android devices" on page 86](#)
- ["Quarantining an Android device based on its OS version" on page 87](#)

Registering/Enrolling a device

Registering and *enrolling* are two terms used to describe the same process of associating a mobile device with a Core mobile device management (MDM) server. An evaluated Common Criteria (CC) configuration allows two enrollment methods for Samsung devices and two for iOS.

- **Samsung devices** can enroll using a provisioning QR code or through the automatic enrollment of a Knox Mobile Enrollment (KME) registered device.

The evaluated Android enrollment methods are described in ["Configuring Knox mobile enrollment" on page 60](#) and ["Provisioning AE devices in a closed network or AOSP deployment" on page 56](#). No other enrollment methods for an Android device are allowed, as they do not meet all CC MDM requirements.

- **iOS devices** can enroll using the MDM Server web enrollment method or through an automatic enrollment of an Apple Device Enrollment Program (DEP) registered device.

The evaluated iOS enrollment methods are described in "Registering iOS and macOS devices through the web" or "Setting up Apple Device Enrollment with MobileIron Core" in *Core Device Management Guide for iOS and macOS Devices*. No other enrollment methods for an iOS device are allowed, as they do not meet all CC MDM requirements.

These enrollment methods utilize a Passcode/PIN which can be set to expire after an given interval. This expiration interval is configurable in **Settings > System Settings > Users & Devices > Registration** in the field **Passcode Expiry**.

Locking a device

To lock a device, see “Lock” in the “Securing Devices” chapter of the *Core Device Management Guide for Android and Android enterprise Devices* and the *Core Device Management Guide for iOS and macOS Devices*.

Wiping a device

To wipe a device (return it to factory defaults) see “Wipe” in the “Securing Devices” chapter of the *Core Device Management Guide for Android and Android enterprise Devices* and the *Core Device Management Guide for iOS and macOS Devices*.

Unenrolling a device

To unenroll (retire) a device, see “Retiring a device” in the “Securing Devices” chapter of the *Core Device Management Guide for Android and Android Enterprise Devices* and the *Core Device Management Guide for iOS and macOS Devices*.

Sending a message to a device

You can send a message to a device or set of devices:

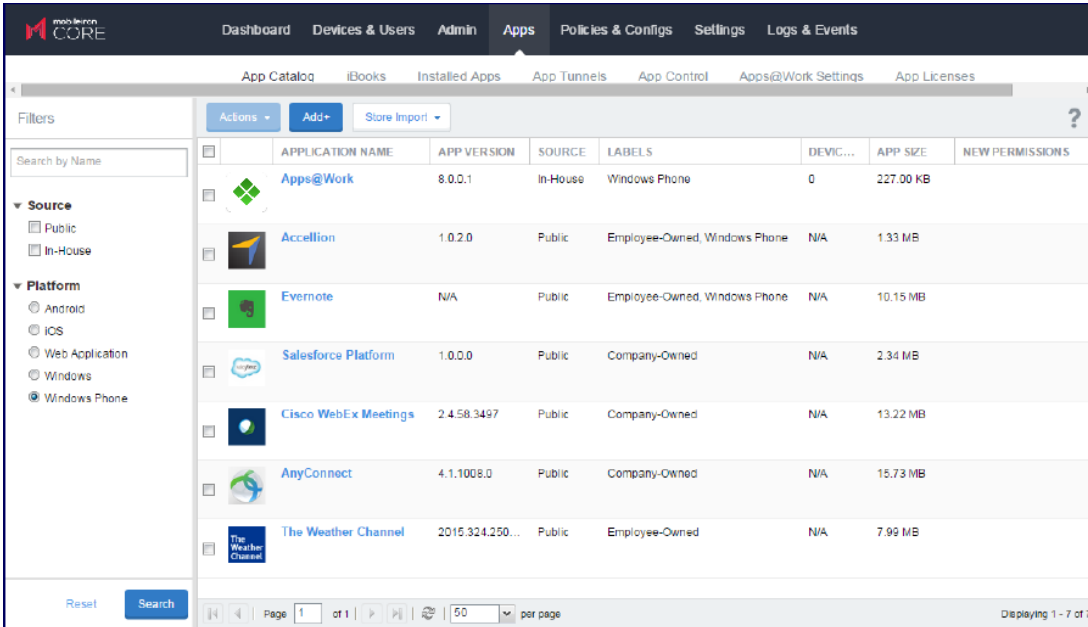
Procedure

1. In the Admin Portal, go to **Devices & Users > Devices**.
2. Select the devices.
3. In the **Send Message** dialog, select how to send the message. Choose one or more of **SMS**, **Email**, **DataChannel**, or **PushNotification**.
4. If you selected **Email**, enter the **Subject**.
5. Enter the message text in the **Message** field.
6. Click **SendMessage**.

The App Catalog

The App Catalog, available from the Admin portal **Apps > App Catalog** is a centralized location for the apps you want to manage for your users. Apps that you add to the Core App Catalog are *managed* apps. By importing apps to the App Catalog, you can make the apps available for users to download to their devices.

FIGURE 1. APP CATALOG



The screenshot displays the MobileIron Core Admin portal's App Catalog. The interface includes a navigation bar with options like Dashboard, Devices & Users, Admin, Apps, Policies & Configs, Settings, and Logs & Events. Below this, there are tabs for App Catalog, iBooks, Installed Apps, App Tunnels, App Control, Apps@Work Settings, and App Licenses. A filters sidebar on the left allows searching by name and filtering by Source (Public, In-House) and Platform (Android, iOS, Web Application, Windows, Windows Phone). The main area shows a table of apps with columns for Application Name, App Version, Source, Labels, Device Count, App Size, and New Permissions. The table lists apps such as Apps@Work, Accellion, Evernote, Salesforce Platform, Cisco WebEx Meetings, AnyConnect, and The Weather Channel.

	APPLICATION NAME	APP VERSION	SOURCE	LABELS	DEVIC...	APP SIZE	NEW PERMISSIONS
<input type="checkbox"/>	Apps@Work	8.0.0.1	In-House	Windows Phone	0	227.00 KB	
<input type="checkbox"/>	Accellion	1.0.2.0	Public	Employee-Owned, Windows Phone	N/A	1.33 MB	
<input type="checkbox"/>	Evernote	N/A	Public	Employee-Owned, Windows Phone	N/A	10.15 MB	
<input type="checkbox"/>	Salesforce Platform	1.0.0.0	Public	Company-Owned	N/A	2.34 MB	
<input type="checkbox"/>	Cisco WebEx Meetings	2.4.58.3497	Public	Company-Owned	N/A	13.22 MB	
<input type="checkbox"/>	AnyConnect	4.1.1008.0	Public	Company-Owned	N/A	15.73 MB	
<input type="checkbox"/>	The Weather Channel	2015.324.250...	Public	Employee-Owned	N/A	7.99 MB	

You use the App Catalog to:

- Add, configure, and remove managed apps
- Install and uninstall managed apps to devices using labels
- Group apps into categories to be displayed in Apps@Work on the device
- Set the prerequisite app for a dependent app
- Indicate mandatory installation of prerequisite apps in Apps@Work
- Use Apple licenses

You can divide these apps into categories of your own definition using *labels*, and take advantage of the additional security features built into AppConnect apps.

You can provide device users with links to recommended iOS apps on the Apple App Store, or links to internally-developed apps they can download from MobileIron Core using Apps@Work on their device.

The App Catalog also allows you to view app details at a glance, such as the app name, size, the version number of in-house apps, the labels to which the app is applied, the origins of the app (public or in-house), and the number of devices to which the app is installed.

An Android in-house app made available through the App Catalog can be designated as a *mandatory* app, which means that the app is always installed on the devices matching the app's labels. An app that is not marked as mandatory is optional, and enables the users to decide whether or not to install the app on their devices. The in-house app can be either an AppConnect app (secure app) or a regular, non-AppConnect app.

Installing and removing apps on a device

After adding any app to the App Catalog, the app must be made available to the relevant users through Apps@Work. This is done by applying the app to a relevant label. The label determines the group of device users who will see the app in Apps@Work on their devices.

Publishing iOS and Android apps to Apps@Work

To publish iOS and Android apps to Apps@Work:

Procedure

1. In the Admin Portal, go to **Apps > App Catalog**.
2. Select **iOS** or **Android** from the Platform list.
3. Select the app you want to work with.
4. Click **Actions > Apply to Label**.
5. Select the label that includes the iOS or Android devices on which you want the selected app to be displayed.
 - In the **Apply to Labels** dialog box, select the check box next to the app name.
 - Click in the **Mandatory** field, a drop-down displays.
 - Selecting **Yes** makes the selected app mandatory; leaving it the default **No** makes the app optional.
6. Click **Apply**.

Collecting, viewing, and exporting logs

Logs relevant to Protection Profile deployments are available and described in the following sections:

- [Viewing audit log information](#)
- ["Exporting Admin Portal audit logs" on the next page](#)
- ["Exporting device status events" on page 123](#)
- ["Collecting audit events for Android devices" on page 126](#)
- ["Exporting System Manager audit logs" on page 135](#)

Viewing audit log information

The Audit Logs page displays the information that MobileIron Core records for your Core instance. You specify what information is displayed on this page when you use the controls in the Filters panel of the page.

To view the information that Core logs:

Procedure

1. In the Admin Portal, go to **Logs**. Core displays the Audit Logs page.

FIGURE 1. AUDIT LOGS PAGE

ACTION	STATE	PERFORMED BY	ACTION DATE	COMPLETED AT	PERFORMED ON	DETAILS
Account Sync Completed	Failed	misystem	2019-05-23 12:35:2...	2019-05-23 12:35:2...	DEP Account	Check update...
Account Sync Completed	Success	misystem	2019-05-23 12:35:2...	2019-05-23 12:35:2...	DEP Account	Check update...
Account Sync Completed	Failed	misystem	2019-05-23 12:20:2...	2019-05-23 12:20:2...	DEP Account	Check update...
Account Sync Completed	Success	misystem	2019-05-23 12:20:2...	2019-05-23 12:20:2...	DEP Account	Check update...
Account Sync Completed	Failed	misystem	2019-05-23 12:05:2...	2019-05-23 12:05:2...	DEP Account	Check update...
Account Sync Completed	Success	misystem	2019-05-23 12:05:2...	2019-05-23 12:05:2...	DEP Account	Check update...
Account Sync Completed	Failed	misystem	2019-05-23 11:50:2...	2019-05-23 11:50:2...	DEP Account	Check update...
Account Sync Completed	Success	misystem	2019-05-23 11:50:2...	2019-05-23 11:50:2...	DEP Account	Check update...
Account Sync Completed	Failed	misystem	2019-05-23 11:35:2...	2019-05-23 11:35:2...	DEP Account	Check update...

The information panel displays:

- **Action** (for example, "Admin Portal" sign-in)
- **State** (for example, "Success")

- **Performed By** (for example, "myadmin")
 - **Action Date**
 - **Completed At**
 - **Performed On** (for example, "Admin Portal")
 - **Details**
2. (Optional) Enter a number in **Page** to specify what page to view.
 3. (Optional) Select a number from **per page** to specify how many records are displayed on a page.
 4. (Optional) Click **Export to CSV** to export the records that match the current search criteria.

Exporting Admin Portal audit logs

You can export Admin portal audit logs from the **Admin portal > Audit Logs** page.

- [Understanding the Admin Portal audit log files](#)
- ["Apply label to policy" audit events](#)
- ["Samples of relevant audit logs" on page 107](#)

To export Admin portal audit logs:

Procedure

1. Navigate to **Admin portal > Audit Logs** page.
2. Click **Export to CSV** and follow the prompts. See ["Viewing audit log information" on the previous page](#) to see the Audit Logs page.

Understanding the Admin Portal audit log files

When you export Admin Portal audit logs to a CSV file, the CSV file has the following comma-separated fields:

TABLE 14. EXPORTED FIELDS IN CSV FILE

Field name	Description
Action	The action the administrator took
State	The status of the action such as succeeded, failed, or initiated
Performed By	User name of the administrator who took the action, or "system" if the Core initiated the action

TABLE 14. EXPORTED FIELDS IN CSV FILE (CONT.)

Field name	Description
Action Date	Time, day, month, and year the action occurred.
Performed On	Component the action is directed to, such as the device, a Core policy or configuration, or LDAP server
Details	Description of the action
Space Name	The delegated administrator space
Space Path	Internal ID of the space
Actor	Same as Performed by
Logged At	Same as Action Date
Version	Not used
User Role	Not used
Object Id	Not used
Object Name	Not used
Subject ID	Internal identifier of user who took the action
Subject Type	Whether the action is on a policy, configuration, or other entity
Subject Owner Name	Not used
Completed At	Same as Logged At
Cookie	Not used
Device	Device details if the action is on a device
Requested	At Time the action was initiated
Configuration	Configuration details if the action is on a configuration
Object Type	Not used
Parent ID	Not used
Update Request ID	Not used
Log Type	Always “userAction”

TABLE 14. EXPORTED FIELDS IN CSV FILE (CONT.)

Field name	Description
Updated Blob	The data of the subject
Message	Not used

"Apply label to policy" audit events

Core generates the "Apply Label to Policy" audit event whenever a policy is applied to a device. One example of this event is shown in ["System Manager audit logs relevant to Protection Profile deployments" on page 138](#). Other examples of this event, triggered by administrator actions, are also relevant to Protection Profile deployments. These other events are not included in this guide due to their large size. However, each one contains configured "ruletype" attributes. These ruletype attributes map directly to the administrator actions.

The following table shows:

- Each of the relevant ruletype attributes
- The related Core policy
- The related administrator action as claimed in the security target, with the corresponding field on the Core policy in parenthesis

TABLE 15. APPLY LABEL TO POLICY AUDIT EVENTS

Ruletype attribute	Core policy	Administrator action (field name on Core policy)
EAS_BLOCK_ANDROID_OS	Security policy	Update system software (Apply compliance action when Android version is less than x)
SECURITY_PWD_TYPE	Security policy	Password policy (Password Type)
SECURITY_PWD_LENGTH	Security policy	Password policy - minimum password length (Minimum Password Length)
SECURITY_PWD_MIN_COMPLEX_CHAR	Security policy	Password policy - Minimum password complexity (Minimum Number of Complex Characters)
SECURITY_PWD_MAX_AGE	Security policy	Password policy - Maximum password lifetime (Maximum Password Age)

TABLE 15. APPLY LABEL TO POLICY AUDIT EVENTS (CONT.)

Ruletype attribute	Core policy	Administrator action (field name on Core policy)
SECURITY_INACTIVITY_TIMEOUT	Security policy	Session locking policy (Maximum Inactivity Timeout)
SECURITY_PWD_MAX_FAILED_ATTEMPTS	Security policy	Session locking policy - number of authentication failures (Maximum Number of Failed Attempts)
LOCKDOWN_WIFI_SSID_LIST	Lockdown policy	Wireless networks (SSIDs) to which the mobile device is not allowed to connect (Turn Off Wi-Fi for SSIDs)
LOCKDOWN_GOOGLE_MARKET	Lockdown policy	Application installation policy (Google Play)
LOCKDOWN_APP_CONTROL_LIST	Lockdown policy	Application installation policy (Restricted Apps, Allowed Apps)
LOCKDOWN_CAMERA	Lockdown policy	Enable/disable policy for camera (Camera)
LOCKDOWN_MICROPHONE	Lockdown policy	Enable/disable policy for microphone (Microphone)
LOCKDOWN_BLUETOOTH	Lockdown policy	Enable/disable policy for Bluetooth (Bluetooth)
LOCKDOWN_NFC	Lockdown policy	Enable/disable policy for NFC (NFC)
LOCKDOWN_WIFI	Lockdown policy	Enable/disable policy for Wi-Fi (Wi-Fi)
LOCKDOWN_CELLULAR_DATA	Lockdown policy	Enable/disable policy for cellular radios (Cellular Data)
LOCKDOWN_WIFI_TETHERING	Lockdown policy	Enable/disable policy for Hotspot (Tethering - Wi-Fi)
LOCKDOWN_BT_TETHERING	Lockdown policy	Enable/disable policy for Bluetooth (Tethering - Bluetooth)

TABLE 15. APPLY LABEL TO POLICY AUDIT EVENTS (CONT.)

Ruletype attribute	Core policy	Administrator action (field name on Core policy)
LOCKDOWN_USB_TETHERING	Lockdown policy	Enable/disable policy for USB (Tethering - USB)
LOCKDOWN_DEVELOPER_OPTIONS	Lockdown policy	Enable/disable policy for developer modes (Developer options)
SECURITY_ENCRYPT_DEVICE	Security policy	Enable policy for data-at-rest protection (Device Encryption)
SECURITY_ENCRYPT_FILE_TYPE	Security policy	Enable policy for data-at-rest protection (File Types)
SECURITY_ENCRYPT_SDCARD	Security policy	Enable policy for removable media's data-at-rest protection (SD Card Encryption)
PRIVACY_ANDROID_LOCK_SCREEN_BANNER_ENABLED	Privacy policy	Configure the unlock banner policy (Enable Warning Banner)
PRIVACY_ANDROID_LOCK_SCREEN_BANNER_TEXT	Privacy policy	Configure the unlock banner policy (Banner Text)
LOCKDOWN_USB_MASS_STORAGE	Lockdown policy	Enable/disable USB mass storage mode (USB Mass Storage)
LOCKDOWN_USB_MEDIA_PLAYER	Lockdown policy	Enable/disable USB mass storage mode (USB Media Player)
LOCKDOWN_GPS	Lockdown policy	Enable/disable location services (GPS)
SECURITY_BLOCK_FINGERPRINT	Security policy	Enable/disable policy for use of Biometric Authentication Factor (Block Fingerprint (from Android 5.0 or Samsung MDM 5.3))
SECURITY_BLOCK_SMART_LOCK	Security policy	Enable/disable policy for use of Biometric Authentication Factor (Block SmartLock (from Android 5.0 only))

Samples of relevant audit logs

The following are examples for logs in the exported Admin Portal Audit log file that are relevant to Protection Profile deployments:

- "Change of policy settings" on the next page
- "Configure security policy for wireless network" on page 109
- "Change in enrollment state (register device) for Android" on page 109
- "Retire Android device" on page 109
- "iOS device registered" on page 109
- "iOS device retired" on page 109
- "iOS device reported error while applying a policy" on page 110
- "Enable/disable policy for the VPN on an iOS device" on page 110
- "Enable/disable policy for display notification in the locked state of all notifications on an iOS device" on page 110
- "Configure the unlock banner policy on an iOS device" on page 111
- "Enable/disable policy for use of Biometric Authentication Factor on an iOS device" on page 111
- "Enable/disable certificate used to validate digital signature on applications on an iOS device" on page 111
- "MDM server start up" on page 111
- "MDM server shutdown" on page 111
- "Add user" on page 112
- "Change user password" on page 112
- "Delete user" on page 112
- "Add label" on page 112
- "Delete label" on page 112
- "Apply label" on page 112
- "Remove label from device" on page 112
- "Activate Policy" on page 113
- "Add Policy" on page 113
- "Delete Policy" on page 114
- "Apply label to policy" on page 115
- "Configure whether users can unenroll from management" on page 116
- "Configure the auditable items on an Android device" on page 116
- "Admin Portal sign in" on page 117
- "Admin Portal sign out" on page 117
- "Modify LDAP settings" on page 118

- ["Add enrollment setting" on page 118](#)
- ["Modify enrollment setting" on page 118](#)
- ["Add trusted certificate" on page 118](#)
- ["Revoke device certificate" on page 118](#)
- ["Force device wakeup" on page 118](#)
- ["Pull client logs" on page 119](#)
- ["Lock device" on page 119](#)
- ["Command Failure " on page 119](#)
- ["Choose X.509v3 certificates for MDM server use" on page 119](#)
- ["Import the certificates to be used for authentication of the MDM Agent communications" on page 120](#)
- ["Install app" on page 120](#)
- ["Remove app" on page 120](#)
- ["Configure app access groups" on page 120](#)
- ["Denying application installation on an iOS device" on page 121](#)
- ["Full wipe of protected data on an Android device" on page 121](#)
- ["Update system software on an iOS device" on page 121](#)
- ["Show banner" on page 121](#)
- ["Hide banner" on page 122](#)
- ["Update banner" on page 122](#)
- ["Update to device limits for registration" on page 122](#)
- ["Configure server session lock timeout" on page 123](#)

Change of policy settings

```
"Modify Policy","Success","admin","2018-10-30 21:24:55 +0000","Samsung General - PolicyName :
Version 2","Policy 'PolicyName' is modified successfully. ","Global","/1/","", "2018-10-30
21:24:55 +0000", "1","", "", "", "", "Policy", "", "2018-10-30 21:24:55 +0000", "", "", "2018-10-30
21:24:55 +0000", "{configId=2, name=PolicyName, configType=SAMSUNG_GENERAL,
version=2}", "", "", "", "userAction", "
{deviceSpaceId:1,deviceSpacePath:""/1/"" ,policyId:2,policyName:""PolicyName"" ,policyVersion:2,
policyType:""ENTERPRISE"" ,profileType:""SAMSUNG_
GENERAL"" ,status:""Active"" ,active:true,defaultPolicy:false,deviceSpaceName:""Global"" ,lastMod
ifiedAt:1540934695581,description:"" <<Description>>"" ,deviceCount:0,pendingCount:0,priority:1,
labels:[],devices:[],mailboxes:
[],deletePolicyFile:false,deleteBooleanFile:false,deleteAuditLogConfigFile:false,rules:
[{ruleType:""SAMSUNG_AUDITING_USERS"" ,value:"" "" ,clientValue:"" "" ,resourceDTOs:[ ]},
{ruleType:""SAMSUNG_FAIL_ATTESTATION_ON_
TIMEOUT"" ,value:"" false"" ,clientValue:"" false"" ,resourceDTOs:[ ]},{ruleType:""SAMSUNG_AUDITING_
GROUPS"" ,value:"" "" ,clientValue:"" "" ,resourceDTOs:[ ]},{ruleType:""SAMSUNG_
ATTESTATION"" ,value:"" false"" ,clientValue:"" false"" ,resourceDTOs:[ ]},{ruleType:""SAMSUNG_
AUDITING_ENABLED"" ,value:"" 2"" ,clientValue:"" 2"" ,resourceDTOs:[ ]},{ruleType:""SAMSUNG_
AUDITING_EVENTS"" ,value:"" "" ,clientValue:"" "" ,resourceDTOs:[ ]},{ruleType:""SAMSUNG_AUDITING_
```

```
SEVERITY_LEVEL"",value:""2"",clientValue:""2"",resourceDTOs:[],{ruleType:""SAMSUNG_KNOX_LICENSE"",value:""<<Knox-license-key>>"",clientValue:""<<Knox-license-key>>"",resourceDTOs:[],{ruleType:""SAMSUNG_AUDITING_OUTCOME_RULE"",value:""2"",clientValue:""2"",resourceDTOs:[],{ruleType:""SAMSUNG_MANAGEMENT_KEY"",resourceDTOs:[],mailboxGuids:[],items:{}}",
```

Configure security policy for wireless network

```
"Apply Label To Configuration","Success","admin","2018-11-05 12:25:35 +0000","WiFi - Action 28 - Wifi Security Policy : Version 1","Label Test Devices applied to configuration Action 28 - Wifi Security Policy","","/1/","","2018-11-05 12:25:35 +0000","1","","3","Test Devices","","Application Setting","","2018-11-05 12:25:35 +0000","","","2018-11-05 12:25:35 +0000","{configId=44, name=Action 28 - Wifi Security Policy, configType=WiFi, version=1}","Label","","","userAction","","",
```

Change in enrollment state (register device) for Android

```
"Register Device","Success","user","2018-10-01 20:54:09 +0000","user (Android 8.0 - PDA 1)","Device is fully registered","","","2018-10-01 20:54:09 +0000","1","","","2e6cda2e-f31a-44e0-8535-70f8968fd86c","Smartphone","","2018-10-01 20:54:09 +0000","","{phoneNumber=PDA 1, uuid=2e6cda2e-f31a-44e0-8535-70f8968fd86c, platform=Android 8.0}","2018-10-01 20:54:09 +0000","","","","userAction","","",
```

Retire Android device

```
"Send Alert","Success","misystem","2018-09-14 20:21:59 +0000","user (Android 8.0 - PDA 5)","(Client #1073741836) was STONITH'd. Reason: rsn mismatch, expected Device is retired, got e46649074d2c1643 and null","","","{principal=user, miUserId=9002, email=gss4testing@gmail.com}","2018-09-14 20:21:59 +0000","1","","","8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136","Smartphone","","2018-09-14 20:21:59 +0000","","{phoneNumber=PDA 5, uuid=8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136, platform=Android 8.0}","2018-09-14 20:21:59 +0000","","","","userAction","","",
```

iOS device registered

```
"Register Device","Success","admin","2018-08-06 16:48:16 +0000","admin (iOS - PDA 2)","Device is fully registered","","","2018-08-06 16:48:16 +0000","1","","","ec7ca012-8fd1-4d68-9d4c-5ee16401b608","Smartphone","","2018-08-06 16:48:16 +0000","","{phoneNumber=PDA 2, uuid=ec7ca012-8fd1-4d68-9d4c-5ee16401b608, platform=iOS}","2018-08-06 16:48:16+0000","","","","userAction","","",
```

iOS device retired

```
"Retire","Success","admin","2018-08-07 10:55:08 +0000","admin (iOS 11.3 - 16505550199)","Request for Retire on the device","Global","/1/","{principal=admin,miUserId=9001, email=gss4testing@gmail.com}","2018-08-07 10:55:08+0000","1","","","ec7ca012-8fd1-4d68-9d4c-5ee16401b608","Smartphone","","2018-08-07 10:55:08 +0000","","{phoneNumber=16505550199, uuid=ec7ca012-8fd1-4d68-9d4c-5ee16401b608, platform=iOS 11.3}","2018-08-07 10:55:08+0000","","","","userAction","","",
```

iOS device reported error while applying a policy

```
"Install Encrypted Sub-Profile","Error",{"MDM"},"2018-10-29 18:21:25 +0000","user(iOS 11.3 - PDA 2)","{"request_data":{"Bad VPN config(VPN.1011)","error":{"<?xml version='1.0'><!DOCTYPE plist SYSTEM'file://localhost/System/Library/DTDs/PropertyList.dtd'><plist version='1.0'><dict><key>CommandUUID</key><string>4ec892fe-bcd6-4686-800f-2af0a8cfb600</string><key>ErrorChain</key><array><dict><key>ErrorCode</key><integer>4001</integer><key>ErrorDomain</key><string>MCInstallationErrorDomain</string><key>LocalizedDescription</key><string>Profile Installation Failed</string><key>USEngishDescription</key><string>Profile Installation Failed</string></dict><dict><key>ErrorCode</key><integer>4001</integer><key>ErrorDomain</key><string>MCInstallationErrorDomain</string><key>LocalizedDescription</key><string>Profile Failed to Install</string><key>USEngishDescription</key><string>Profile Failed to Install</string></dict><dict><key>ErrorCode</key><integer>1009</integer><key>ErrorDomain</key><string>MCProfileErrorDomain</string><key>LocalizedDescription</key><string>The profile &#8220;Bad VPN config&#8221; could not be installed.</string><key>USEngishDescription</key><string>The profile &#8220;BadVPN config&#8221; could not be installed.</string></dict><dict><key>ErrorCode</key><integer>15000</integer><key>ErrorDomain</key><string>MCVPNErrrorDomain</string><key>LocalizedDescription</key><string>PPTP is deprecated starting from iOS 10 and valid configurations can not becreated.</string><key>USEngishDescription</key><string>PPTP is deprecated starting from iOS 10 and valid configurations can not be created.</string></dict></array><key>Status</key><string>Error</string><key>UDID</key><string>019037a6715246588d1c36cea8ef0a1c2303e952</string></dict></plist></n","message":"Completed MDM request."},"","","{principal=user, miUserId=null, email=user},"2018-10-29 18:21:25+0000","1","","11","Bad VPN config (VPN.1011)","f5eb0f65-5251-4bff-b8ec-9bb7c01c09d9","MDM Event","","2018-10-29 18:21:25 +0000","","{phoneNumber=PDA 2,uuid=f5eb0f65-5251-4bff-b8ec-9bb7c01c09d9, platform=iOS 11.3},"2018-10-29 18:21:17+0000","","Application Setting","","","userAction",,"
```

Enable/disable policy for the VPN on an iOS device

```
"Apply Label To Configuration","Success","admin","2018-11-17 20:11:16 +0000","Restrictions - Action 31 - VPN Protection : Version 1","Label Test Devices applied to configuration Action 31 - VPN Protection","","/1/","","2018-11-17 20:11:16 +0000","1","","3","Test Devices","","Application Setting","","2018-11-17 20:11:16 +0000","","","2018-11-17 20:11:16 +0000","{configId=14, name=Action 31 - VPN Protection, configType=Restrictions, version=1},"Label","","","userAction",,"
```

Enable/disable policy for display notification in the locked state of all notifications on an iOS device

```
"Apply Label To Configuration","Success","admin","2018-11-17 20:31:01 +0000","Restrictions - Action 40 - Notifications : Version 1","Label Test Devices applied to configuration Action 40 - Notifications","","/1/","","2018-11-17 20:31:01 +0000","1","","3","Test Devices","","Application Setting","","2018-11-17 20:31:01 +0000","","","2018-11-17 20:31:01
```

```
+0000", "{configId=15, name=Action 40 - Notifications, configType=Restrictions, version=1}", "Label", "", "", "userAction", "",
```

Configure the unlock banner policy on an iOS device

```
"Settings", "Acknowledged", "<MDM>", "2018-08-08 12:21:17 +0000", "user (iOS 11.3 - 16505550199)", "{ \"request_data\": \"Wallpaper={\\n \\\"referenceCountProperty\\\" : \\\"476a90b6-d88d-4ae5-ab88-2e79f4752a9e\\\", \\n \\\"wallpaperImageInfoList\\\" : [ {\\n \\\"wallpaperScreenImageType\\\" : \\\"LOCK_SCREEN\\\", \\n \\\"fileId\\\" : \\\"8f5c6bcd-fb66-4cbb-8943-a10edbf4f45\\\" } ]\\n }\", \"message\": \"Completed MDM request.\"\", \"\", \"\", \"\", \"principal=user, miUserId=null, email=user\", \"2018-08-08 12:21:17 +0000\", \"1\", \"\", \"\", \"476a90b6-d88d-4ae5-ab88-2e79f4752a9e\", \"wallpaperImageInfoList\" \"\", \"Wallpaper={ \"referenceCountProperty\" : \"476a90b6-d88d-4ae5-ab88-2e79f4752a9e\", \"wallpaperImageInfoList\" : [ { \"wallpaperScreenImageType\" : \"LOCK_SCREEN\", \"fileId\" : \"8f5c6bcd-fb66-4cbb-8943-a10edbf4f45\" } ]\", \"1edbd995-4057-42e3-a4bb-a37e8764ac49\", \"MDM Event\", \"\", \"2018-08-08 12:21:17 +0000\", \"\", \"{phoneNumber=16505550199, uuid=1edbd995-4057-42e3-a4bb-a37e8764ac49, platform=iOS 11.3}\", \"2018-08-08 12:21:10 +0000\", \"\", \"Application Setting\", \"\", \"\", \"userAction\", \"\",
```

Enable/disable policy for use of Biometric Authentication Factor on an iOS device

```
"Apply Label To Configuration", "Success", "admin", "2018-11-17 20:22:23 +0000", "Restrictions - Action 55 - Bio Auth : Version 1", "Label Test Devices applied to configuration Action 55 - Bio Auth", \"\", \"/1/\", \"\", \"2018-11-17 20:22:23 +0000\", \"1\", \"\", \"3\", \"Test Devices\", \"\", \"Application Setting\", \"\", \"2018-11-17 20:22:23 +0000\", \"\", \"\", \"2018-11-17 20:22:23 +0000\", \"{configId=17, name=Action 55 - Bio Auth, configType=Restrictions, version=1}\", \"Label\", \"\", \"\", \"userAction\", \"\",
```

Enable/disable certificate used to validate digital signature on applications on an iOS device

```
"Apply Label To Configuration", "Success", "admin", "2018-08-06 15:22:05 +0000", "Provisioning Profile - ios 2aeb0df6-c861-4a24-91b7-e8671d707a78 : Version 1", "Label iOS applied to configuration ios 2aeb0df6-c861-4a24-91b7-e8671d707a78\", \"\", \"/1/\", \"\", \"2018-08-06 15:22:05 +0000\", \"1\", \"\", \"-4\", \"iOS\", \"\", \"Application Setting\", \"\", \"2018-08-06 15:22:05 +0000\", \"\", \"\", \"2018-08-06 15:22:05 +0000\", \"{configId=37, name=ios 2aeb0df6-c861-4a24-91b7-e8671d707a78, configType=Provisioning Profile, version=1}\", \"Label\", \"\", \"\", \"userAction\", \"\",
```

MDM server start up

```
"Application Started", "Success", "misystem", "2018-07-25 15:31:30 +0000", "System", "Core application started\", \"\", \"\", \"\", \"2018-07-25 15:31:30 +0000\", \"1\", \"\", \"AuditLog Service\", \"\", \"\", \"NA\", \"\", \"2018-07-25 15:31:30 +0000\", \"\", \"\", \"2018-07-25 15:31:30 +0000\", \"\", \"\", \"\", \"\", \"userAction\", \"\",
```

MDM server shutdown

```
"Application Stopped", "Initiated", "misystem", "2018-07-25 15:33:00 +0000", "System", "Core application shutdown initiated\", \"\", \"\", \"\", \"2018-07-25 15:33:00 +0000\", \"1\", \"\", \"AuditLog Service\", \"\", \"\", \"NA\", \"\", \"2018-07-25 15:33:00 +0000\", \"\", \"\", \"2018-07-25 15:33:00 +0000\", \"\", \"\", \"\", \"\", \"userAction\", \"\",
```



```
of apps removed: 0","Global","/1/","{principal=user, miUserId=9002, email=null}","2018-09-14
19:56:06 +0000","1","","3","Test Device","aaca17ea-8754-403a-b964-
4b2f0e2aba66","Smartphone","","2018-09-14 19:56:06 +0000","","{phoneNumber=PDA 2,
uuid=aaca17ea-8754-403a-b964-4b2f0e2aba66, platform=Android 8.0}","2018-09-14 19:56:06
+0000","","Label","","","userAction", ""
```

Activate Policy

```
"Activate Policy","Success","gssadmin","2021-04-30 15:49:49 +0000","Device Name - Action 39b
iOS Change Name : Version 1","Policy 'Action 39b iOS Change Name' is activated successfully.
","Global","/1/","","2021-04-30 15:49:49 +0000","1","","","","","Policy","","2021-04-30
15:49:49 +0000","","","2021-04-30 15:49:49 +0000","{configId=8, name=Action 39b iOS Change
Name, configType=APPLE_DEVICE_NAME, version=1}","","","","userAction", "
{deviceSpaceId:1,deviceSpacePath:""/1/""",policyId:8,policyName:""Action 39b iOS Change
Name""",policyVersion:1,policyType:""ENTERPRISE""",profileType:""APPLE_DEVICE_
NAME""",status:""Active""",active:true,defaultPolicy:false,deviceSpaceName:""Global""",lastModifi
edAt:1619797789852,description:""GSS Action 39b Test
Profile""",deviceCount:0,pendingCount:0,priority:1,policyIdForPriority:0,labels:[],devices:
[],mailboxes:
[],deletePolicyFile:false,deleteBooleanFile:false,deleteAuditLogConfigFile:false,rules:
[{ruleType:""DEVICE_NAME""",value:""GSS Test Device""",clientValue:""GSS Test
Device""",resourceDTOs:[]}],mailboxGuids:[],items:{}}",
```

Add Policy

```
"Add Policy","Success","miadmin","2018-10-22 18:44:33 +0000","Sync - SyncTest : Version
1","Policy 'SyncTest' is added successfully. ","Global","/1/","","2018-10-22 18:44:33
+0000","1","","","","","Policy","","2018-10-22 18:44:33 +0000","","","2018-10-22 18:44:33
+0000","{configId=2, name=SyncTest, configType=SYNC, version=1}","","","","userAction", "
{deviceSpaceId:1,deviceSpacePath:""/1/""",policyId:2,policyName:""SyncTest""",policyVersion:1,pol
icyType:""ENTERPRISE""",profileType:""SYNC""",status:""Active""",active:true,defaultPolicy:false
,deviceSpaceName:""Global""",lastModifiedAt:1540233873111,deviceCount:0,pendingCount:0,priority
:1,labels:[],devices:[],mailboxes:
[],deletePolicyFile:false,deleteBooleanFile:false,deleteAuditLogConfigFile:false,rules:
[{ruleType:""SYNC_LONGER_INTERVAL""",value:""42""",clientValue:""42""",resourceDTOs:[]},
{ruleType:""SYNC_HEARTBEAT_INTERVAL""",value:""14""",clientValue:""840""",resourceDTOs:[]},
{ruleType:""SYNC_MULTITASK_INVERVAL""",value:""15""",clientValue:""15""",resourceDTOs:[]},
{ruleType:""SYNC_REQUIRE_TLS""",value:""on""",clientValue:""yes""",resourceDTOs:[]},
{ruleType:""SYNC_OS_UPDATE_URL""",resourceDTOs:[]}, {ruleType:""MIGRATE_
CLIENT""",value:""off""",clientValue:""no""",resourceDTOs:[]}, {ruleType:""SYNC_
INTERVAL""",value:""240""",clientValue:""14400""",resourceDTOs:[]}, {ruleType:""SYNC_NTP_
SERVER""",resourceDTOs:[]}, {ruleType:""PUSH_NOTIFICATION_
MECHANISM""",value:""auto""",clientValue:""auto""",resourceDTOs:[]}, {ruleType:""SYNC_FULL_BG_
MODE""",value:""off""",clientValue:""off""",resourceDTOs:[]}, {ruleType:""SYNC_OS_UPDATE_
NOW""",resourceDTOs:[]}, {ruleType:""MA_CERT_RENEWAL_
WINDOW""",value:""60""",clientValue:""60""",resourceDTOs:[]}, {ruleType:""SYNC_
SERVERIP""",value:""app283.auto.mobileiron.com""",clientValue:""app283.auto.mobileiron.com""",res
ourceDTOs:[]}, {ruleType:""SYNC_BLOCK_WHEN_
ROAMING""",value:""mai""",clientValue:""on""",resourceDTOs:[]}, {ruleType:""SYNC_MTD_WAKEUP_
INTERVAL""",value:""15""",clientValue:""900""",resourceDTOs:[]}, {ruleType:""SYNC_OS_UPDATE_
SCHED""",resourceDTOs:[]}, {ruleType:""SYNC_ALWAYS_
```

CONNECTED",value:"off",clientValue:"off",resourceDTOs:[],{ruleType:"MA_CERT_GRACE_PERIOD",value:"30",clientValue:"30",resourceDTOs:[]},mailboxGuids:[],items:{}}",

Delete Policy

"Delete Policy","Success","gssadmin","2021-02-18 18:21:09 +0000","Lockdown - Action 11 - Android 5.0 : Version 1","Policy 'Action 11 - Android 5.0' is deleted successfully.", "Global", "/1/", "", "2021-02-18 18:21:09 +0000", "1", "", "", "", "", "Policy", "", "2021-02-18 18:21:09 +0000", "", "", "2021-02-18 18:21:09 +0000", "{configId=14, name=Action 11 - Android 5.0, configType=LOCKDOWN, version=1}", "", "", "", "userAction", "{deviceSpaceId:1,deviceSpacePath:""/1/"",policyId:14,policyName:""Action 11 - Android 5.0"",policyVersion:1,policyType:""ENTERPRISE"",profileType:""LOCKDOWN"",status:""Active"",active:true,defaultPolicy:false,deviceSpaceName:""Global"",lastModifiedAt:1613669219048,deviceCount:0,pendingCount:0,priority:5,policyIdForPriority:0,labels:[],devices:[],mailboxes:[],deletePolicyFile:false,deleteBooleanFile:false,deleteAuditLogConfigFile:false,rules:[{ruleType:""LOCKDOWN_AFW_BLUETOOTH_CONTACT_SHARING"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_NFC"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_GPS"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_WP_ALLOW_MANUAL_MDM_UNENROLLMENT"",value:""on"",clientValue:""on"",resourceDTOs:[],{ruleType:""LOCKDOWN_AFW_BLUETOOTH_CONFIG"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_AFW_SCREEN_CAPTURE"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_MANUAL_WIFI_SETUP"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_USB_DEBUG"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_SAMSUNG_GOOGLE_PLAY"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_ALWAYS_CONNECT_MANAGED_WIFI"",value:""off"",clientValue:""1"",resourceDTOs:[],{ruleType:""LOCKDOWN_WP_ALLOW_PRINTING"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_HEALTH_DEVICE"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_CERT_REVOCATION_STATUS"",value:""off"",clientValue:""1"",resourceDTOs:[],{ruleType:""LOCKDOWN_ENABLE_SAMSUNG_RESTRICTIONS"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_AFW_VPN_CONFIG"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_AFW_CONTACT_SEARCH"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_AFW_WHITELISTED_ACCESSIBILITY_SERVICES"",value:""[]"",clientValue:""[]"",resourceDTOs:[],{ruleType:""LOCKDOWN_ROAMING_VOICE_CALLS"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_SEBOOLEAN_FILE_CHUNK_COUNT"",value:""0"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_SAMSUNG_OUTGOING_MMS"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_ALLOW_USB_HID_PROTOCOL"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_BT_TETHERING"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_ALLOW_BROWSER_POPUPS"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_ALLOW_BROWSER_PASSWORD_MANAGER"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_WP_ALLOW_AUTO_FILL"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_AFW_KEEP_ON"",value:""off"",clientValue:""1"",resourceDTOs:[],{ruleType:""LOCKDOWN_SAFE_SEARCH"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_SAMSUNG_ALLOWED_APPS_LIST_COUNT"",value:""0"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_SAMSUNG_BLUETOOTH_TETHERING"",value:""on"",clientValue:""0"",resourceDTOs:[],{ruleType:""LOCKDOWN_SAMSUNG_WIFI_SSID_LIST_HASH"",value:""E3B0C44298FC1C149AFBF4C8996FB92427AE41E4649B934CA495991B7852B855"",clientValue:""E3B0C44298FC1C149AFBF4C8996FB92427AE41E4649B934CA495991B7852B855"",resourceDTOs:[],{ruleType:""LOCKDOWN_POLICYZIP_FILENAME"",value:""",clientValue:""",resourceDTOs:[],{ruleType:""LOCKDOWN_WIFI_TETHERING"",value:""on"",clientValue:""0"",resourceDTOs:[],

```
{ruleType:""LOCKDOWN_AFW_ALLOW_CAMERA"",value:""on"",clientValue:""0"",resourceDTOs:[]},
{ruleType:""LOCKDOWN_SAMSUNG_FACTORY_RESET"",value:""on"",clientValue:""0"",resourceDTOs:[]},
{ruleType:""LOCKDOWN_SAMSUNG_USB_MEDIA_PLAYER"",value:""on"",clientValue:""0"",resourceDTOs:
[]},{ruleType:""LOCKDOWN_AFW_WIFI_SLEEP_CONFIG"",value:""on"",clientValue:""0"",resourceDTOs:
[]},{ruleType:""LOCKDOWN_GOOGLE_BACKUP"",value:""on"",clientValue:""0"",resourceDTOs:[]},
{ruleType:""LOCKDOWN_WP_ALLOW_USER_TO_RESET_
PHONE"",value:""on"",clientValue:""on"",resourceDTOs:[]}, {ruleType:""LOCKDOWN_INTERNET_
SHARING"",value:""on"",clientValue:""0"",resourceDTOs:[]}, {ruleType:""LOCKDOWN_SEBOOLEAN_
FILENAME"",value:""","",clientValue:""","",resourceDTOs:[]}, {ruleType:""LOCKDOWN_AFW_
WIFI"",value:""on"",clientValue:""0"",resourceDTOs:[]}, {ruleType:""LOCKDOWN_SAMSUNG_WIFI_
TETHERING"",value:""on"",clientValue:""0"",resourceDTOs:[]}, {ruleType:""LOCKDOWN_WP_ALLOW_
SHARE_INTO_PROFILE"",value:""on"",clientValue:""0"}
```

Apply label to policy



This example also shows the event logged when changing the sync interval, which is the periodicity for when Core receives device information.

The “Apply label to policy” event is also logged for other administrator actions, detailed in [“Apply label to policy” audit events](#) on page 104

```
"Apply Label To Policy","Success","miadmin","2018-10-22 18:56:21 +0000","Sync - SyncTest :
Version 1","Label 'Android' is applied to policy 'SyncTest'.","Global","/1/","", "2018-10-22
18:56:21 +0000","1","", "-10","Android","", "Policy","", "2018-10-22 18:56:21
+0000","", "", "2018-10-22 18:56:21 +0000","{configId=2, name=SyncTest, configType=SYNC,
version=1}", "LABEL","", "", "userAction", "
{deviceSpaceId:1,deviceSpacePath:""/1/"",policyId:2,policyName:""SyncTest"",policyVersion:1,po
licyType:""ENTERPRISE"",profileType:""SYNC"",status:""Active"",active:true,defaultPolicy:false
,deviceSpaceName:""Global"",lastModifiedAt:1540233873111,deviceCount:0,pendingCount:0,priority
:1,labels:[],devices:[],mailboxes:
[],deletePolicyFile:false,deleteBooleanFile:false,deleteAuditLogConfigFile:false,rules:
[{ruleType:""SYNC_LONGER_INTERVAL"",value:""42"",clientValue:""42"",resourceDTOs:[]},
{ruleType:""SYNC_HEARTBEAT_INTERVAL"",value:""14"",clientValue:""840"",resourceDTOs:[]},
{ruleType:""SYNC_MULTITASK_INVERVAL"",value:""15"",clientValue:""15"",resourceDTOs:[]},
{ruleType:""SYNC_REQUIRE_TLS"",value:""on"",clientValue:""yes"",resourceDTOs:[]},
{ruleType:""SYNC_OS_UPDATE_URL"",resourceDTOs:[]}, {ruleType:""MIGRATE_
CLIENT"",value:""off"",clientValue:""no"",resourceDTOs:[]}, {ruleType:""SYNC_
INTERVAL"",value:""240"",clientValue:""14400"",resourceDTOs:[]}, {ruleType:""SYNC_NTP_
SERVER"",resourceDTOs:[]}, {ruleType:""PUSH_NOTIFICATION_
MECHANISM"",value:""auto"",clientValue:""auto"",resourceDTOs:[]}, {ruleType:""SYNC_FULL_BG_
MODE"",value:""off"",clientValue:""off"",resourceDTOs:[]}, {ruleType:""SYNC_OS_UPDATE_
NOW"",resourceDTOs:[]}, {ruleType:""MA_CERT_RENEWAL_
WINDOW"",value:""60"",clientValue:""60"",resourceDTOs:[]}, {ruleType:""SYNC_
SERVERIP"",value:""app283.auto.mobileiron.com"",clientValue:""app283.auto.mobileiron.com"",res
ourceDTOs:[]}, {ruleType:""SYNC_BLOCK_WHEN_
ROAMING"",value:""mai"",clientValue:""on"",resourceDTOs:[]}, {ruleType:""SYNC_MTD_WAKEUP_
INTERVAL"",value:""15"",clientValue:""900"",resourceDTOs:[]}, {ruleType:""SYNC_OS_UPDATE_
SCHED"",resourceDTOs:[]}, {ruleType:""SYNC_ALWAYS_
CONNECTED"",value:""off"",clientValue:""off"",resourceDTOs:[]}, {ruleType:""MA_CERT_GRACE_
PERIOD"",value:""30"",clientValue:""30"",resourceDTOs:[]},mailboxGuids:[],items:{"}}
```

Configure whether users can unenroll from management

```
"Apply Label To Policy","Success","admin","2018-10-31 22:33:44 +0000","Lockdown - Prevent Unenroll Policy : Version 1","Label 'Test Devices' is applied to policy 'Prevent Unenroll Policy' .","Global","/1/","", "2018-10-31 22:33:44 +0000","1","", "3","Test Devices","", "Policy","", "2018-10-31 22:33:44 +0000","", "2018-10-31 22:33:44 +0000", "{configId=4, name=Prevent Unenroll Policy, configType=LOCKDOWN, version=1}", "LABEL","", "userAction", "{deviceSpaceId:1,deviceSpacePath:""/1/",policyId:4,policyName:""Prevent Unenroll Policy"",policyVersion:1,policyType:""ENTERPRISE"",profileType:""LOCKDOWN"",status:""Active"", active:true,defaultPolicy:false,deviceSpaceName:""Global"",lastModifiedAt:1541025212976,deviceCount:0,pendingCount:0,priority:1,labels:[],devices:[],mailboxes:[],deletePolicyFile:false,deleteBooleanFile:false,deleteAuditLogConfigFile:false,rules:[...,{ruleType:""LOCKDOWN_MANAGEMENT_REMOVAL"",value:""off"",clientValue:""1"",resourceDTOs:[],...},mailboxGuids:[],items:{}}",
```



Note that because of the size of this audit record, parts of the lockdown policy that contain settings that are not relevant to this audit are not displayed in this sample message. These other settings reflect attributes that have the potential to change depending on the lockdown policy configured by the administrator. However, the audit record will show at least the information shown in this sample message.

Configure the auditable items on an Android device

```
"Apply Label To Policy","Success","admin","2018-08-23 10:14:46 +0000","Samsung General - Samsung Audit Config : Version 1","Label 'Android' is applied to policy 'Samsung Audit Config' .","Global","/1/","", "2018-08-23 10:14:46 +0000","1","", "10","Android","", "Policy","", "2018-08-23 10:14:46 +0000","", "2018-08-23 10:14:46 +0000", "{configId=27, name=Samsung Audit Config, configType=SAMSUNG_GENERAL, version=1}", "LABEL","", "userAction", "{deviceSpaceId:1,deviceSpacePath:""/1/",policyId:27,policyName:""Samsung Audit Config"",policyVersion:1,policyType:""ENTERPRISE"",profileType:""SAMSUNG_GENERAL"",status:""Active"", active:true,defaultPolicy:false,deviceSpaceName:""Global"",lastModifiedAt:1535018868040,deviceCount:0,pendingCount:0,priority:1,labels:[],devices:[],mailboxes:[],deletePolicyFile:false,deleteBooleanFile:false,deleteAuditLogConfigFile:false,rules:[{ruleType:""SAMSUNG_AUDITING_USERS"",value:""",clientValue:"",resourceDTOs:[]}, {ruleType:""SAMSUNG_MANAGEMENT_KEY"",resourceDTOs:[]}, {ruleType:""SAMSUNG_AUDITING_ENABLED"",value:""1"",clientValue:""1"",resourceDTOs:[]}, {ruleType:""SAMSUNG_FAIL_ATTESTATION_ON_TIMEOUT"",value:""false"",clientValue:""false"",resourceDTOs:[]}, {ruleType:""SAMSUNG_ATTESTATION"",value:""false"",clientValue:""false"",resourceDTOs:[]}, {ruleType:""SAMSUNG_AUDITING_OUTCOME_RULE"",value:""2"",clientValue:""2"",resourceDTOs:[]}, {ruleType:""SAMSUNG_KNOX_LICENSE"",value:""KLM06-D6ZNW-W7K42-P307T-IFJD1-F4FHO"",clientValue:""KLM06-D6ZNW-W7K42-P307T-IFJD1-F4FHO"",resourceDTOs:[]}, {ruleType:""SAMSUNG_AUDITING_GROUPS"",value:""0,1,2,3,4"",clientValue:""0,1,2,3,4"",resourceDTOs:[]}, {ruleType:""SAMSUNG_AUDITING_SEVERITY_LEVEL"",value:""4"",clientValue:""4"",resourceDTOs:[]}, {ruleType:""SAMSUNG_AUDITING_EVENTS"",value:""0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29"",clientValue:""0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29"",resourceDTOs:[]},mailboxGuids:[],items:{}}",
```

Apply label to configuration

```
"Apply Label To Configuration","Success","admin","2018-10-29 18:20:34 +0000","VPN - Bad VPN
config : Version 1","Label Test Devices applied to configuration Bad VPN
config",",","/1/",",",",2018-10-29 18:20:34 +0000","1",",",3","Test Devices",",",Application
Setting",",",2018-10-29 18:20:34 +0000",",",",2018-10-29 18:20:34 +0000","{configId=11,
name=Bad VPN config, configType=VPN, version=1}","Label",",",",",userAction",",",
```

Remove label from configuration

```
"Remove Label From Configuration","Success","admin","2018-11-05 12:20:53 +0000","Certificate -
Action 9 RootCA ECDSA : Version 1","Label Test Devices removed from configuration Action 9
RootCA ECDSA",",","/1/",",",",2018-11-05 12:20:53 +0000","1",",",3","Test
Devices",",",Application Setting",",",2018-11-05 12:20:53 +0000",",",",2018-11-05 12:20:53
+0000","{configId=11, name=Action 9 RootCA ECDSA, configType=Certificate,
version=1}","Label",",",",",userAction",",",
```

Prevent unenrollment on Android devices



This event is an "Apply Label To Policy" event for the Lockdown policy when the field **Management Removal** is disabled.

```
"Apply Label To Policy","Success","admin","2018-10-31 22:33:44 +0000","Lockdown - Prevent
Unenroll Policy : Version 1","Label 'Test Devices' is applied to policy 'Prevent Unenroll
Policy' .","Global","/1/",",",2018-10-31 22:33:44 +0000","1",",",3","Test
Devices",",",Policy",",",2018-10-31 22:33:44 +0000",",",",2018-10-31 22:33:44 +0000",
{configId=4, name=Prevent Unenroll Policy, configType=LOCKDOWN,
version=1}","LABEL",",",",",userAction",
{deviceSpaceId:1,deviceSpacePath:""/1/"",policyId:4,policyName:""Prevent Unenroll
Policy""",policyVersion:1,policyType:""ENTERPRISE""",profileType:""LOCKDOWN""",status:""Active""",
active:true,defaultPolicy:false,deviceSpaceName:""Global""",lastModifiedAt:1541025212976,device
Count:0,pendingCount:0,priority:1,labels:[],devices:[],mailboxes:
[],deletePolicyFile:false,deleteBooleanFile:false,deleteAuditLogConfigFile:false,rules:[...
,{ruleType:""LOCKDOWN_MANAGEMENT_REMOVAL""",value:""off""",clientValue:""1""",resourceDTOs:
[],...}],mailboxGuids:[],items:{}}",
```

Admin Portal sign in

```
"Admin Portal Sign In","Success","admin","2018-09-13 15:28:50 +0000","Admin Portal -
10.0.0.130","Successfully Signed In",",",",",2018-09-13 15:28:50
+0000","1",",",",",",",Admin Portal",",",2018-09-13 15:28:50 +0000",",",",2018-09-13
15:28:50 +0000",",",",",",userAction",",",
```

Admin Portal sign out

```
"Admin Portal Sign Out","Success","admin","2018-09-14 17:42:26 +0000","Admin Portal -
10.0.0.209","Successfully Signed Out",",",",",2018-09-14 17:42:26
+0000","1",",",",",",",Admin Portal",",",2018-09-14 17:42:26 +0000",",",",2018-09-14
17:42:26 +0000",",",",",",userAction",",",
```

Modify LDAP settings

```
"Modify LDAP","Success","admin","2018-09-13 19:55:13 +0000","LDAP - ldaps://tlv0-16x.example.com:636","LDAP Setting is modified for the server ldaps://tlv0-16x.example.com:636","","","2018-09-13 19:55:13 +0000","1","","","","LDAP","","2018-09-13 19:55:13 +0000","","","2018-09-13 19:55:13 +0000","{configId=null, name=ldaps://tlv0-16x.example.com:636, configType=LDAP, version=null}","","","userAction",",",
```

Add enrollment setting

```
"Add Configuration","Success","admin","2018-09-13 15:41:33 +0000","SCEP - SubSubCA-RSA Enrollment Cert : Version 1","Configuration SubSubCA-RSA Enrollment Cert added","","/1/","","2018-09-13 15:41:33 +0000","1","","","","Application Setting","","2018-09-13 15:41:33 +0000","","","2018-09-13 15:41:33 +0000","{configId=11, name=SubSubCA-RSA Enrollment Cert, configType=SCEP, version=1}","","","userAction",",",
```

Modify enrollment setting

```
"Modify Configuration","Success","admin","2018-09-13 15:41:47 +0000","SCEP - Root-RSA Enrollment Cert : Version 2","Configuration Root-RSA Enrollment Cert modified","","/1/","","2018-09-13 15:41:47 +0000","1","","","","Application Setting","","2018-09-13 15:41:47 +0000","","","2018-09-13 15:41:47 +0000","{configId=10, name=Root-RSA Enrollment Cert, configType=SCEP, version=2}","","","userAction",",",
```

Add trusted certificate

```
"Modify Configuration","Success","misystem","2018-09-13 15:17:30 +0000","Certificate - System - TLS Trust Certificate Chain for Mobile Management : Version 3","Configuration System - TLS Trust Certificate Chain for Mobile Management modified","","/1/","","2018-09-13 15:17:30 +0000","1","","","","Application Setting","","2018-09-13 15:17:30 +0000","","","2018-09-13 15:17:30 +0000","{configId=-4, name= System - TLS Trust Certificate Chain for Mobile Management, configType=Certificate, version=3}","","","userAction",",",
```

Revoke device certificate

```
"Revoke Device Certificate","Success","admin","2018-09-14 20:21:48 +0000","user (Android 8.0 - PDA 5)"," SCEP Name: 'System - Mutual Auth CE setting', Serial Number: '106',","","","{principal=user, miUserId=9002, email=gss4testing@gmail.com}","2018-09-14 20:21:48 +0000","1","","6a","6a","8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136","Certificate","","2018-09-14 20:21:48 +0000","","","{phoneNumber=PDA 5, uuid=8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136, platform=Android 8.0}","2018-09-14 20:21:48 +0000","","Certificate","","userAction",",",
```

Force device wakeup

```
"Wakeup","Success","admin","2018-09-14 20:21:23 +0000","user (Android 8.0 - PDA 5)","Request for Force Device Check-In.,"Global","/1/","{principal=user, miUserId=9002, email=gss4testing@gmail.com}","2018-09-14 20:21:23 +0000","1","","","8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136","Smartphone","","2018-09-14 20:21:23 +0000","","","{phoneNumber=PDA 5, uuid=8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136, platform=Android 8.0}","2018-09-14 20:21:23 +0000","","","userAction",",",
```


Pull client logs

```
"Pull Client Logs","Success","System","2018-09-14 20:21:35 +0000","user (Android 8.0 - PDA 5)","Successfully uploaded client logs","","","{principal=user, miUserId=9002, email=}","2018-09-14 20:21:35 +0000","1","","","","8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136","Smartphone","","2018-09-14 20:21:35 +0000","","{phoneNumber=PDA 5, uuid=8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136, platform=Android 8.0}","2018-09-14 20:21:35 +0000","","","","","userAction",,""
```

Lock device

```
"Send Message","Initiated","misystem","2018-09-14 20:19:58 +0000","user (Android 8.0 - PDA 5)","Auto-generated message to notify action: Lock via email","","","{principal=user, miUserId=null, email=gss4testing@gmail.com}","2018-09-14 20:19:58 +0000","1","","","","Smartphone","","2018-09-14 20:19:58 +0000","","{phoneNumber=PDA 5, uuid=null, platform=Android 8.0}","2018-09-14 20:19:58 +0000","","","","","userAction",,""
```

```
"Lock","Success","admin","2018-09-14 20:19:58 +0000","user (Android 8.0 - PDA 5)","Request for Lock on the device","Global","/1/","{principal=user, miUserId=9002, email=gss4testing@gmail.com}","2018-09-14 20:19:58 +0000","1","","","","8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136","Smartphone","","2018-09-14 20:19:58 +0000","","{phoneNumber=PDA 5, uuid=8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136, platform=Android 8.0}","2018-09-14 20:19:58 +0000","","","","","userAction",,""
```

Command Failure

(from **Admin Portal (Devices & Users > Devices > Actions)**)

```
"Send Message","Failed","admin","2018-09-14 20:20:44 +0000","user (Android 8.0 - PDA 5)","Request from Admin Portal to deliver via gcm - Failure Message: Failed to dispatch notification to MI Gateway. Reason: Internal Error.','','{principal=user, miUserId=9002, email=gss4testing@gmail.com}","2018-09-14 20:20:44 +0000","1","","","","8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136","Smartphone","","2018-09-14 20:20:44 +0000","","{phoneNumber=PDA 5, uuid=8e7b1fa0-dbb5-41e0-aaad-d9f9a5fe8136, platform=Android 8.0}","2018-09-14 20:20:44 +0000","","","","","userAction",,""
```

Choose X.509v3 certificates for MDM server use

```
"Add Configuration","Success","admin","2018-09-13 15:41:33 +0000","SCEP - SubSubCA-RSA Enrollment Cert : Version 1","Configuration SubSubCA-RSA Enrollment Cert added","","/1/","","2018-09-13 15:41:33 +0000","1","","","","Application Setting","","2018-09-13 15:41:33 +0000","","","2018-09-13 15:41:33 +0000","{configId=11, name=SubSubCA-RSA Enrollment Cert, configType=SCEP, version=1}","","","","userAction",,""
```

```
"Modify Configuration","Success","misystem","2018-09-13 15:17:30 +0000","Certificate - System - TLS Trust Certificate Chain for Mobile Management : Version 3","Configuration System - TLS Trust Certificate Chain for Mobile Management modified","","/1/","","2018-09-13 15:17:30 +0000","1","","","","Application Setting","","2018-09-13 15:17:30 +0000","","","2018-09-13 15:17:30 +0000","{configId=-4, name= System - TLS Trust Certificate Chain for Mobile Management, configType=Certificate, version=3}","","","","userAction",,""
```

Import the certificates to be used for authentication of the MDM Agent communications

```
"Create Device Certificate","Success","misystem","2018-10-29 16:24:58 +0000","user (Android 8.0 - PDA 3)"," SCEP Name: 'System - Mutual Auth CE setting', Serial Number: '101',",,","", "{principal=user, miUserId=9002, email=gss4testing2@gmail.com}", "2018-10-29 16:24:58 +0000", "1", "", "65", "65", "87022827-1865-47e5-8b2a-2f36d3738e93", "Certificate", "", "2018-10-29 16:24:58 +0000", "", "{phoneNumber=PDA 3, uuid=87022827-1865-47e5-8b2a-2f36d3738e93, platform=Android 8.0}", "2018-10-29 16:24:58 +0000", "", "Certificate", "", "", "userAction", "", "Register Device","Success","user","2018-10-29 16:24:58 +0000","user (Android 8.0 - PDA 3)","Device is fully registered","", "", "", "2018-10-29 16:24:58 +0000", "1", "", "", "", "87022827-1865-47e5-8b2a-2f36d3738e93", "Smartphone", "", "2018-10-29 16:24:58 +0000", "", "{phoneNumber=PDA 3, uuid=87022827-1865-47e5-8b2a-2f36d3738e93, platform=Android 8.0}", "2018-10-29 16:24:58 +0000", "", "", "", "", "userAction", "",
```

Install app

```
"Add App","Success","admin","2018-10-29 18:02:34 +0000","Android Utility 1.0 (Android - In-House)","Application 'Android Utility 1.0' supported on Android platform is added to catalog","Global", "/1/", "", "2018-10-29 18:02:34 +0000", "1", "", "", "", "118", "App", "", "2018-10-29 18:02:34 +0000", "", "", "2018-10-29 18:02:34 +0000", "", "", "", "", "userAction", "", "Apply Label To App","Success","admin","2018-10-29 18:03:00 +0000","Android Utility 1.0 (Android - In-House)","Label 'Test Devices' is applied to application 'Android Utility'.","Global", "/1/", "", "2018-10-29 18:03:00 +0000", "1", "", "3", "Test Devices", "118", "App", "", "2018-10-29 18:03:00 +0000", "", "", "2018-10-29 18:03:00 +0000", "", "LABEL", "", "", "userAction", "", "Install App","Success","user","2018-10-29 18:05:18 +0000","user (Android 8.0 - PDA 3)","App Android Utility 1.0 Installed","", "", "{principal=user, miUserId=9002, email=}", "2018-10-29 18:05:18 +0000", "1", "", "", "", "87022827-1865-47e5-8b2a-2f36d3738e93", "Smartphone", "", "2018-10-29 18:05:18 +0000", "", "{phoneNumber=PDA 3, uuid=87022827-1865-47e5-8b2a-2f36d3738e93, platform=Android 8.0}", "2018-10-29 18:05:18 +0000", "", "", "", "", "userAction", "",
```

Remove app

```
Remove Label From App","Success","admin","2018-10-30 21:34:34 +0000","Android Utility 1.0 (Android - In-House)","Label 'Test Devices' is removed from application 'Android Utility'.","Global", "/1/", "", "2018-10-30 21:34:34 +0000", "1", "", "3", "Test Devices", "118", "App", "", "2018-10-30 21:34:34 +0000", "", "", "2018-10-30 21:34:34 +0000", "", "LABEL", "", "", "userAction", "", Uninstall App","Success","user","2018-10-31 16:24:50 +0000","user (Android 8.0 - PDA 3)","App Android Utility 1.0 Uninstalled","", "", "{principal=user, miUserId=9002, email=}", "2018-10-31 16:24:50 +0000", "1", "", "", "", "87022827-1865-47e5-8b2a-2f36d3738e93", "Smartphone", "", "2018-10-31 16:24:50 +0000", "", "{phoneNumber=PDA 3, uuid=87022827-1865-47e5-8b2a-2f36d3738e93, platform=Android 8.0}", "2018-10-31 16:24:50 +0000", "", "", "", "", "userAction", "",
```

Configure app access groups

```
"Apply Label To App","Success","admin","2018-10-29 18:03:00 +0000","Android Utility 1.0 (Android - In-House)","Label 'Test Devices' is applied to application 'Android
```



```
Utility'.", "Global", "/1/", "", "2018-10-29 18:03:00 +0000", "1", "", "3", "Test  
Devices", "118", "App", "", "2018-10-29 18:03:00 +0000", "", "", "2018-10-29 18:03:00  
+0000", "", "LABEL", "", "", "userAction", "",
```

Denying application installation on an iOS device

```
"Apply Label To Configuration", "Success", "admin", "2018-11-17 19:56:38 +0000", "Restrictions -  
Action 29 - Deny App Installation : Version 1", "Label Test Devices applied to configuration  
Action 29 - Deny App Installation", "", "/1/", "", "2018-11-17 19:56:38 +0000", "1", "", "3", "Test  
Devices", "", "Application Setting", "", "2018-11-17 19:56:38 +0000", "", "", "2018-11-17 19:56:38  
+0000", "{configId=13, name=Action 29 - Deny App Installation, configType=Restrictions,  
version=1}", "Label", "", "", "userAction", "",
```

Full wipe of protected data on an Android device

```
Wipe", "Success", "admin", "2018-08-15 11:06:25 +0000", "user (Android 8.0 - PDA 18)", "Request for  
Wipe on the device", "Global", "/1/", "{principal=user, miUserId=9002,  
email=johnmessiha@gossamersec.com}", "2018-08-15 11:06:25 +0000", "1", "", "", "", "c1c4fcb1-eef5-  
4d03-88b4-868260d19191", "Smartphone", "", "2018-08-15 11:06:25 +0000", "", "{phoneNumber=PDA 18,  
uuid=c1c4fcb1-eef5-4d03-88b4-868260d19191, platform=Android 8.0}", "2018-08-15 11:06:25  
+0000", "", "", "", "", "userAction", "",
```

Full wipe of protected data on an iOS device

```
"Wipe", "Success", "admin", "2018-08-15 15:43:57 +0000", "user (iOS 11.3 - 16505550199)", "Request  
for Wipe on the device", "Global", "/1/", "{principal=user, miUserId=9002,  
email=johnmessiha@gossamersec.com}", "2018-08-15 15:43:57 +0000", "1", "", "", "", "1edbd995-4057-  
42e3-a4bb-a37e8764ac49", "Smartphone", "", "2018-08-15 15:43:57 +0000", "", "  
{phoneNumber=16505550199, uuid=1edbd995-4057-42e3-a4bb-a37e8764ac49, platform=iOS  
11.3}", "2018-08-15 15:43:57 +0000", "", "", "", "", "userAction", "",
```

Update system software on an iOS device

```
"Update OS Software", "Success", "admin", "2018-08-17 12:35:52 +0000", "user (iOS 11.3 - PDA  
40)", "Request for Update OS Software on the device", "Global", "/1/", "{principal=user,  
miUserId=9002, email=johnmessiha@gossamersec.com}", "2018-08-17 12:35:52  
+0000", "1", "", "", "", "62a28583-7570-4724-976b-dfb07986b1c1", "Smartphone", "", "2018-08-17  
12:35:52 +0000", "", "{phoneNumber=PDA 40, uuid=62a28583-7570-4724-976b-dfb079
```

Show banner



On the Admin Portal, at **Logs > Audit Logs**, this event appears in the category **Other** with the event name **Preference Config Changes**.

```
"Preference Config Changes", "Success", "admin", "2018-10-01 22:05:23 +0000", "System", "Show Login  
banner", "", "", "", "2018-10-01 22:05:23 +0000", "1", "", "", "", "Settings Preferences", "", "2018-  
10-01 22:05:23 +0000", "", "", "2018-10-01 22:05:23 +0000", "", "", "", "userAction", "",
```

Hide banner



On the Admin Portal, at **Logs > Audit Logs**, this event appears in the category **Other** with the event name **Preference Config Changes**.

```
"Preference Config Changes","Success","admin","2018-10-01 22:06:19 +0000","System","Hide Login banner","","","","2018-10-01 22:06:19 +0000","1","","","","","Settings Preferences","","2018-10-01 22:06:19 +0000","","","2018-10-01 22:06:19 +0000","","","","","userAction",,""
```

Update banner



On the Admin Portal, at **Logs > Audit Logs**, this event appears in the category **Other** with the event name **Preference Config Changes**.

```
"Preference Config Changes","Success","miadmin","2021-07-15 20:49:14 +0000","System","Update Login banner text","","","","2021-07-15 20:49:14 +0000","1","","","","","Settings Preferences","","2021-07-15 20:49:14 +0000","","","2021-07-15 20:49:14 +0000","","","","","userAction",,""
```

Update to device limits for registration



On the Admin Portal, at **Logs > Audit Logs**, these events appear in the category **Other** with the event name **Preference Config Changes**.

```
"Preference Config Changes","Success","misystem","2018-10-31 21:08:44 +0000","System","Modify Preference limitDevices to 2","","","","2018-10-31 21:08:44 +0000","1","","","","","Settings Preferences","","2018-10-31 21:08:44 +0000","","","2018-10-31 21:08:44 +0000","","","","","userAction",,""
```

Update to the client identify certificate for mutual authentication



On the Admin Portal, at **Logs > Audit Logs**, these events appear in the category **Other** with the event name **Preference Config Changes**.

When mutual authentication has been enabled with the default certificate enrollment setting in which Core is the local Certificate Authority:

```
"Preference Config Changes","Success","misystem","2018-10-30 00:14:47 +0000","System","Modify Preference enableClientCertAuth from false to true","","","","2018-10-30 00:14:47 +0000","1","","","","","Settings Preferences","","2018-10-30 00:14:47 +0000","","","2018-10-30 00:14:47 +0000","","","","","userAction",,""
```

When the selected certificate enrollment setting for the client identity certificate for mutual authentication has been changed from the default to a SCEP certificate enrollment setting. The numbers -8 and 10 are setting IDs that are internal to Core.

```
"Preference Config Changes", "Success", "misystem", "2018-11-02 00:12:33 +0000", "System", "Modify Preference clientCertAuthSCEPSettingId from -8 to 10", "", "", "", "2018-11-02 00:12:33 +0000", "1", "", "", "", "", "Settings Preferences", "", "2018-11-02 00:12:33 +0000", "", "", "2018-11-02 00:12:33 +0000", "", "", "", "", "userAction", "",
```

Configure server session lock timeout



On the Admin Portal, at **Logs > Audit Logs**, these events appear in the category **Other** with the event name **Preference Config Changes**.

```
"Preference Config Changes", "Success", "misystem", "2018-10-30 14:36:58 +0000", "System", "Modify Preference adminPortalSessionTimeout from 60 to 5", "", "", "", "2018-10-30 14:36:58 +0000", "1", "", "", "", "", "Settings Preferences", "", "2018-10-30 14:36:58 +0000", "", "", "2018-10-30 14:36:58 +0000", "", "", "", "", "userAction", "",
```

Exporting device status events

- ["Device status events overview" below](#)
- ["Creating a device status event" on the next page](#)
- ["Configuring email settings in the System Manager" on the next page](#)
- ["Exporting the device status events as a CSV file" on the next page](#)
- ["Samples of relevant audit logs" on page 107](#)

Device status events overview

Core logs device status events. Device status events alert the administrator when a change on a device indicates a possible security issue, such as when the user attempts to unenroll. Several of these events are relevant to Protection Profile deployments of Android devices. You can export these events to a CSV file.

Creating a device status event

To create a device status event, in the Admin Portal:

Procedure

1. Go to **Logs > Event Settings**.
2. Click **Add New**.
3. Select **Device Status Event** from the dropdown menu.
4. Select these options:
 - a. **Device status is changed**
 - b. **Android device reports policy/config errors** (applicable only to Android devices)
 - c. **Android device reports policy/config warnings** (applicable only to Android devices)
5. In **Actions**, in **Alert Configuration**, for **Severity**, select **Warning**.
6. Click **Save**.

Configuring email settings in the System Manager

When creating a device status event, you can choose to alert an administrator with an email. For Core to send this email, configure the email settings in the System Manager at **Settings > Email Settings**.

Exporting the device status events as a CSV file

To export device status events into a comma separated values (CSV) file:

Procedure

1. In the Admin Portal, go to **Logs > Events**.
2. For **Type**, select **Device Status Alert**.
3. Click **Export**. A CSV file downloads to your computer.

The exported device status event CSV file contains the following fields:

TABLE 16. DEVICE STATUS EVENT CSV FILE FIELDS

Field	Description
Alert Type	The type of alert which in this case is “device status event”
Alert Status	The status of the alert, such as whether it has been dispatched
Comments	The optional note added by the administrator before generating the CSV file
Alert Text	The text of the alert
Recipient	Either user, admin, or both
Alert Date	When the alert was sent
IsActive	Always true
Device UUID	UUID of the device causing the event
Alert Severity	Critical, warning, or information

Samples of relevant device status events

The following events in the exported CSV file are relevant to Protection Profile deployments:

TABLE 17. DEVICE STATUS EVENTS RELEVANT TO PROTECTION PROFILE DEPLOYMENTS

Event	Sample Message
Android device registered	"DEVICE_STATUS_ALERT", "DISPATCHED", "", "WARNING::PDA 11 (user) Device Registered", "user, []", "Tue Aug 21 16:16:01 UTC 2018", "TRUE", "c152d1f1-5a95-4cc4-af25-360049b0251f", "WARNING"
Android device retired	"DEVICE_STATUS_ALERT", "DISPATCHED", "", "WARNING::PDA 10 (user) Device RETIRE initiated", "user, []", "Tue Aug 21 16:16:01 UTC 2018", "TRUE", "dcee86ef-420c-4b72-be7e-e399d41b816c", "WARNING"
Android device reported warning while applying a policy	"DEVICE_STATUS_ALERT", "DISPATCHED", "", "WARNING::PDA 6 (user) Device reported warning(s) while applying the following policies: Lockdown.", "user, []", "Tue Sep 18 16:11:01 UTC 2018", "TRUE", "0dde8523-ce4d-44ce-a135-1ca8b588c337", "WARNING"
Android device reported error while applying a policy	"DEVICE_STATUS_ALERT", "DISPATCHED", "", "WARNING::PDA 6 (user) Device reported error(s) while applying the following configurations: Name: FAU_ALT_EXT.1-t2-unsupported-vpn & Type: VPN.", "user, []", "Tue Sep 18 16:11:01 UTC 2018", "TRUE", "0dde8523-ce4d-44ce-a135-1ca8b588c337", "WARNING"

Collecting audit events for Android devices

- ["Overview of audit events on Samsung Knox devices" below](#)
- ["Format of events generated on Samsung Knox devices" on the next page](#)
- ["Events generated on Samsung Knox devices" on page 128](#)
- ["Events generated on Samsung Knox devices due to changes to the lockdown policy" on page 132](#)
- ["Payload signature events generated on Samsung Knox devices" on page 133](#)
- ["Pulling device logs to Core" on page 134](#)
- ["Accessing audits and device logs on Core" on page 134](#)

Overview of audit events on Samsung Knox devices

The Samsung General Policy provides audit collection control settings. These settings control what audit events are logged to the device logs on Samsung Knox devices based on an event's severity, outcome, and audit group. With some exceptions, these settings impact all logs collected on the Samsung device: logs made by the Samsung platform, as well as logs made by Mobile@Work. The exceptions are the logs in the log.txt file, described in ["Payload signature events generated on Samsung Knox devices" on page 133](#).

You pull these device logs to Core, and then can access them using the System Manager.

Configuring the Samsung General Policy

Configure the audit collection control settings on the Samsung General Policy to control what audit events are logged to the device logs on Samsung Knox devices based on an event's severity, outcome, and audit group.

Procedure

1. In the Admin Portal, go to **Policies & Configs > Policies**.
2. Select the Samsung General Policy that you are using.
3. Click **Edit**,
4. For **Audit Collection Controls**, select **Enable**.

5. For **Severity Rule**, select the severity level of events you want to collect. Only audit events of the chosen severity level or higher will be collected. For example, if you select **Error**, only **Error**, **Critical**, and **Alert** audit events will be collected. The severity levels are, from most severe to least severe, are:
 - Alert
 - Critical
 - Error (the default)
 - Warning
 - Notice
6. For **Outcome Rule**, select whether you want to collect only events indicating success, events indicating failure, or all.
7. For **Audit Groups**, select the groups of events you want to collect. Select one or more of **Security**, **System**, **Network**, **Events**, or **Application**. The default is that all of the groups are selected.
8. If you selected **Events** in the **Audit Groups** field, the **Audit Events** field is enabled. The possible individual events in the **Events** group are displayed in the **Audit Events** dropdown. Select the individual events that you want to collect. For descriptions of the events, see "[Format of events generated on Samsung Knox devices](#)" below.
9. In the **UID** section, click the + sign to add a UID. Each UID is an integer, defined by Samsung, for enabling Samsung-specific logging.
10. Click **Save**.

Format of events generated on Samsung Knox devices

The events in the following tables are generated on Samsung devices:

- "[Events generated on Samsung Knox devices](#)" on the next page
- "[Format of events generated on Samsung Knox devices](#)" above
- "[Payload signature events generated on Samsung Knox devices](#)" on page 133

Example

"[Events generated on Samsung Knox devices](#)" on the next page and "[Events generated on Samsung Knox devices due to changes to the lockdown policy](#)" on page 132 have the format as given in the following example:

```
1531304770883 5/4/1/21636/0/com.mobileiron.mdmp/SamsungLockdownProvider/  
ENABLE_DISABLE_MIC : Disabled
```

Where:

- **1531304770883** - Timestamp of the event occurrence.
- **5** - Severity
- **4** - Module group
- **1** - Outcome. 1 for success and 0 for failure
- **21636** - Process ID (PID) that triggered the event
- **0** - User ID that triggered the event
- **com.mobileiron.mdmpp** - Package name of Mobile@Work for Android in most cases, except
- **SamsungLockdownProvider** - Software component where the event occurred
- **ENABLE_DISABLE_MIC** : - The event name
- **Disabled** - Details about the event

For brevity, the **Message** column of the table contains only the event name and details about the event.

Events generated on Samsung Knox devices

Except where noted in the table, these events belong to the **Events** audit group. They are logged when the Samsung General Policy is configured as follows:

- **Audit Collection Controls** is enabled.
- **Events** is selected in the **Audit Groups** field.
- The specific event is selected in the **Audit Events** field.
- The **Severity** level of the event is not more severe than the selected level in the **Severity Rule** field.
- The outcome of the event matches the selection in the **Outcome Rule** field.

The format of the events are described in "[Format of events generated on Samsung Knox devices](#)" on the [previous page](#).

TABLE 18. EVENTS GENERATED ON SAMSUNG KNOX DEVICES

Event	Description of when the event is logged	Message	Severity
Audit configuration change	A change in audit configuration (which is part of Samsung General Policy)	<p>AUDIT_CONFIGURATION_CHANGE: AuditLog configuration changed: <code>_description_of_change_</code></p> <hr/> <p>The value <code>_description_of_change_</code> shows the current severity rule, current outcome rule, and the list of active audit groups, audit events, and UID list after the change.</p> <hr/> <p>For example:</p> <pre>severityRule = 5 outcomeRule = 2 groupsRule = [4] usersList = [0] events: [AUDIT_CONFIGURATION_CHANGE, POLICY_APPLICATION]</pre>	Alert
Transition to locked state	Device becomes locked either due to user action on the device or administrator action on Core.	TRANSITION_TO_LOCKED_STATE : Screen is off	Notice
Policy validation failure	Finding a discrepancy in policy validation.	POLICY_VALIDATION_FAILURE: <code>_description_of_policy_error_</code>	Error for policy errors. Warning for policy warnings.
CA certificate import	Installation (success or failure) of CA certificates.	CA_CERTIFICATE_IMPORT: Import of CA certificate <code>_alias_</code> succeeded/failed	Notice if success. Error if failure.
ID certificate import	Installation (success or failure) of identity certificates.	ID_CERTIFICATE_IMPORT: Import of ID certificate <code>_alias_</code> succeeded/failed	Notice if success. Error if failure

TABLE 18. EVENTS GENERATED ON SAMSUNG KNOX DEVICES (CONT.)

Event	Description of when the event is logged	Message	Severity
CA certificate remove	Removal (success or failure) of CA certificates.	CA_CERTIFICATE_REMOVE: Removal of CA certificate _alias_ succeeded/failed	Notice if success. Error if failure.
ID certificate remove	Removal (success or failure) of identity certificates.	ID_CERTIFICATE_REMOVE: Removal of ID certificate _alias_ succeeded/failed	Notice
Install application	Installation of an application.	INSTALL_APPLICATION: _package_name_	Notice
Remove application	Removal of an application.	REMOVE_APPLICATION: _package_name_	Notice
Device OS upgrade	Device system software upgrade.	SYSTEM_SOFTWARE_UPGRADE: _current_os_version_	Notice
Server request client checkin	Logs wakeup command from Core that results in a device checkin from Mobile@Work.	WAKEUP_FROM_SERVER: CheckIn with server initiated	Notice
Request current software version	Attempt to fetch and report current software version.	QUERY_CURRENT_SOFTWARE_VERSION: _OS_VERSION_	Notice
Request current hardware version	Attempt to fetch and report current hardware version.	QUERY_CURRENT_HARDWARE_VERSION: _device_model_	Notice
Report app inventory	Attempt to report apps inventory.	REPORT_APP_INVENTORY: Reporting of app inventory succeeded/failed.	Notice if success. Error if failure
Upload logs	Attempt to upload logs to server.	UPLOAD_LOGS: Client logs uploaded/Client logs upload failed: _error_code_	Notice if success. Error if failure.

TABLE 18. EVENTS GENERATED ON SAMSUNG KNOX DEVICES (CONT.)

Event	Description of when the event is logged	Message	Severity
Configure App Store quarantine compliance action	Apps@Work (in Mobile@Work) is quarantined or taken out of quarantine.	APP_STORE_QUARANTINE_ACTION_CONFIGURED: Quarantined/Unquarantined	Notice
Enable/Disable camera	Enabling/disabling camera.	ENABLE_DISABLE_CAMERA: Enabled/Disabled	Notice
Enable/Disable mic	Enabling/disabling microphone.	ENABLE_DISABLE_MIC: Enabled/Disabled	Notice
Enable/Disable radios	Enabling/disabling any of Wi-Fi, NFC, cellular, or Bluetooth.	ENABLE_DISABLE_RADIOS: NFC/WIFI/Bluetooth enabled/disabled	Notice
Enable/Disable development mode	Enabling/disabling development mode.	ENABLE_DISABLE_DEVELOPMENT_MODE: Enabled/Disabled	Notice
Configure unlock banner	Changing and applying unlock banner policy.	CONFIGURE_UNLOCK_BANNER: Enabled/Disabled	Notice if success. Error if failure.
Enable/Disable usb tether	Enabling/disabling of USB tethering.	ENABLE_DISABLE_USB_TETHER: Enabled/Disabled	Notice
Enable/Disable location services	Enabling/disabling of location services.	ENABLE_DISABLE_LOCATION_SERVICES: Enabled/Disabled	Notice
Enable/Disable biometrics	Enabling/disabling biometric authentication.	ENABLE_DISABLE_BIOMETRICS: Fingerprint/Iris/Face unlock enabled/enabling failed.	Notice if success. Error if failure.
Failure to establish a TLS session	Failure in establishing TLS session, including noting any certificate error.	TLS_SESSION_FAILURE: _description_of_failure_reason	Warning
TLS session established	Established TLS session with Core.	TLS_SESSION_ESTABLISHED:	Notice

TABLE 18. EVENTS GENERATED ON SAMSUNG KNOX DEVICES (CONT.)

Event	Description of when the event is logged	Message	Severity
TLS session terminated	Terminated TLS session with Core	TLS_SESSION_TERMINATED:	Warning
Failure to verify presented identity	Failure to verify presented identity	TLS_SESSION_FAILURE : The server connection is rejected because an unsafe SSL certificate is detected. Details: Hostname mismatch. <MobileIron Core FQDN> != <hostname in the certificate>	Warning
Android shutdown	Android is shutting down. This event belongs to the System audit group and is logged if the System group is selected in the Audit Groups field of the Samsung General Policy These events are logged by the Samsung device rather than Mobile@Work so the format is different.	1541176318823 5/2/1/3377/-1/ShutdownThread/Android will be shutdown	Notice
Android startup completed	Android startup has completed. This event belongs to the System audit group and is logged if the System group is selected in the Audit Groups field of the Samsung General Policy These events are logged by the Samsung device rather than Mobile@Work so the format is different.	1541176421450 5/2/1/3379/-1/ActivityManagerService/Android boot completed	Notice

Events generated on Samsung Knox devices due to changes to the lockdown policy

Changes to the lockdown policy that is applied to the device result in the events in the following table. These events:

- Are logged by Mobile@Work, except where noted.
- Are part of the audit group “Events,” except as noted.
- Are logged according to the audit collection controls on the Samsung General Policy.

- The format of the events are described in ["Format of events generated on Samsung Knox devices" on page 127](#).

Payload signature events generated on Samsung Knox devices

Mobile@Work for Android logs payload signature events on Samsung Knox devices, but these events are not controlled by the audit collection controls on the Samsung General Policy. These events can be pulled to Core and are available in the log.txt file, as described in ["Pulling device logs to Core" on the next page](#) and ["Accessing audits and device logs on Core" on the next page](#).

The payload signature events have the following format:

```
<date><time>:<log level>:<process ID>:<thread name>:<class name>:<message>
```

where:

- <date> is the date the event was logged
- <time> is the time the event was logged
- <log level> is a one letter code indicating the log level which for these events is either **I** for Info or **W** for Warning.
- <process ID> indicates the process Mobile@Work that logged the event
- <class name> indicates the class name of the code in Mobile@Work that logged the event
- <message> depends on the event

The following table shows only the <class name> and <message>.



The audit records shown in [Table 19](#) are always recorded, regardless of the selection criteria currently being enforced.

TABLE 19. PAYLOAD SIGNATURE EVENTS GENERATED ON SAMSUNG KNOX DEVICES

Event	Description of when the event is logged	Class name and Message	Log Level
Valid policy signature	The payload has a valid signature.	PayloadSignatureVerifier:Payload signature found	Info
		PayloadSignatureVerifier:Payload signature verified with cert: <certificate>	Info

TABLE 19. PAYLOAD SIGNATURE EVENTS GENERATED ON SAMSUNG KNOX DEVICES (CONT.)

Event	Description of when the event is logged	Class name and Message	Log Level
Incorrectly signed policy	Signature verification for payload failed.	PayloadSignatureVerifier: Payload signature found	Info
		PayloadSignatureVerifier: Payload signature verification failed with cert: <certificate>	Warning
Corrupted signature for policy	The payload signature is corrupted.	PayloadSignatureVerifier:Payload signature found	Info
		PayloadSignatureVerifier: Payload signature mismatch	Warning

Pulling device logs to Core

You can pull device logs from a device to which a Samsung General Policy has been applied. The device logs are pulled to Core. The device logs include the `log.txt` file and the `dump.gz` file. To export these logs, see ["Accessing audits and device logs on Core" below](#).

Procedure

1. In the Admin Portal, go to **Devices & Users > Devices**.
2. Select the check box next to a device. You may only select one device.
3. Click the **Actions** button.
4. Select **Android only > Pull Client Logs**.

The **Pull Client Logs** screen is displayed. The **Devices** field displays the name of the device that you selected. You cannot add additional devices here.

5. Click the **Pull Client Logs** button. If the device is active, it will send the logs to Core. Otherwise, it will send the logs on its next device check-in.

Accessing audits and device logs on Core

The System Manager interface is used to pull device logs from agents and to export System Manager audit logs. Use the following instructions to obtain an archive file containing the audit logs and device logs.

Procedure

1. In a browser, go to `https://<fully_qualified_hostname>:8443/mics`.
2. Enter the user ID and password of a System Manager user.
3. In the System Manager, go to the **Troubleshooting** tab.
4. In the section **Export Logs**, select **Show Tech (All Logs)**.
5. In the **Export Type** field, select **Download**.
6. Click the **Export (Download)** button. Core downloads an archive file (tgz file) containing the following files:
 - MICS.log
 - MIFS.log
 - Application.log
 - /var/log/httpd/https-error_log
 - /var/log/httpd/portal_error_log
 - /var/log/secure
 - upgrade.log.

In the archive file, navigate to this location to locate device logs pulled from agents:

- `<showtechall directory>/log/tomcat/clientlogs/<timestamp>/dump.gz`
The file `dump.gz` is an archive file containing the log files from the device. The timestamp is when the `dump.gz` file was created.
- `<showtechall directory>/log/tomcat/clientlogs/<timestamp>/log.txt`
The `log.txt` file contains payload signature events from the device. The timestamp is when the `log.txt` file was created.

Exporting System Manager audit logs

System Manager audit logs provide many logs that are relevant to Protection Profile deployments. The following topics provide an overview of System Manager audit logs.

- **To export System Manager audit logs**, see ["Accessing audits and device logs on Core" on the previous page](#)

- **To understand the format of event messages**, see the following topics:
 - ["Format of events in MICS log, MIFS log, and Application logs"](#) below
 - ["Format of events in /var/log/httpd/https-error_log and /var/log/httpd/portal_error_log"](#) below
 - ["Format of events in /var/log/secure"](#) on the next page
 - ["Format of events in upgrade.log"](#) on the next page
- **To see examples of audit log messages**, see ["System Manager audit logs relevant to Protection Profile deployments"](#) on page 138

Format of events in MICS log, MIFS log, and Application logs

Events in the MICS log, MIFS, log, and Application log have the following format:

```
<date> <timestamp> <log level> <class name and method> <thread ID> <message>
```

where:

- **<date>** is the date the event was logged.
- **<timestamp>** is the time the event was logged.
- **<log level>** is either ERROR, WARNING, INFO, DEBUG, TRACE.
- **<class name and method>** indicates the code in Core that logged the event.
- **<thread ID>** indicates the thread in Core that logged the event.
- **<message>** depends on the event. The **<message>** is shown in ["System Manager audit logs relevant to Protection Profile deployments"](#) on page 138.

Example

```
2018-11-01 21:15:00,024 INFO [DataLogger.log] (MIReportScheduler_Worker-1:)
t=MDM_APNS_CERTIFICATE_EXPIRY_WARNING status=UNKNOWN NUM_OF_DAYS=12
```

Format of events in /var/log/httpd/https-error_log and /var/log/httpd/portal_error_log

Events in `/var/log/httpd/https-error_log` and `/var/log/httpd/portal_error_log` files have the following format:

```
<timestamp> <module>:<log level> <process ID> <IP address of client> <log message>
```

where:

- **<timestamp>** is the time the event was logged
- **<module>** is the code module in Core that logged the event

- `<log level>` is either ERROR, WARNING, INFO, DEBUG, TRACE
- `<process ID>` indicates the process in Core that logged the event
- `<IP address of client>` indicates the IP address of the client making the request
- `<message>` depends on the event. The `<message>` is shown in ["System Manager audit logs relevant to Protection Profile deployments" on the next page.](#)

Apache logs these errors. Apache uses the following format for the above fields:

```
ErrorLogFormat "[%{u}t] [%-m:%l] [pid %P:tid %T] [client\ %a] %M%
```

See <https://httpd.apache.org/docs/2.4/mod/core.html#errorlogformat>.

Format of events in /var/log/secure

Events in log file /var/log/secure have the following format:

```
<timestamp> <hostname> <module name> <message>
```

where:

- `<timestamp>` is the time the event was logged
- `<hostname>` is the hostname of Core
- `<module name>` indicates the code module in Core that logged the event
- `<message>` depends on the event.

All of these event fields are shown in ["System Manager audit logs relevant to Protection Profile deployments" on the next page.](#)

Example

```
Oct 23 15:15:50 localhost useradd[7004]: new user: name=miadmin, UID=2002, GID=1001, home=/mi, shell=/mobileiron.com/programs/com.mobileiron.core.base/bin/clish
```

Format of events in upgrade.log

Events in upgrade.log have the following format:

```
<timestamp> <hostname> <subject> <message>
```

where:

- `<timestamp>` is the time the event was logged
- `<hostname>` is the host name of Core.
- `<subject>` is the identity of the user responsible for the activity
- `<message>` depends on the event

All of these event fields are shown in "[System Manager audit logs relevant to Protection Profile deployments](#)" below.

System Manager audit logs relevant to Protection Profile deployments

The following audit log events relevant to Protection Profile deployments are available in the System Manager in the Troubleshooting tab:

- **MIFS log messages**

- "[Failure of key generation activity for authentication key](#)" on the next page
- "[Failure of the randomization process](#) " on the next page
- "[Failure to establish a TLS session](#) " on the next page
- "[Failure to verify presented identifier](#) " on the next page
- "[Establishment of a TLS session](#)" on the next page
- "[Termination of a TLS session](#)" on page 140
- "[Failure to validate X.509 certificate – Expired Certificate](#)" on page 140
- "[Failure to validate X.509 certificate – Revoked Certificate](#)" on page 140
- "[Failure to validate X.509 certificate – Revocation Server no CRLsign](#)" on page 140
- "[Failure to validate X.509 certificate –Missing BasicConstraints](#)" on page 140
- "[Failure to validate X.509 certificate – CA Flag is False](#)" on page 141
- "[Failure to validate X.509 certificate – No KeyCertSign KeyUsage](#)" on page 141
- "[Failure to validate X.509 certificate – Bad Path Length](#)" on page 141
- "[Failure to validate X.509 certificate – Explicit curve in certificate](#)" on page 142
- "[Failure to establish connection to determine revocation status.](#) " on page 142
- "[Failed user login during enrollment](#)" on page 143
- "[Enrollment attempted after expiration of authentication data](#) " on page 143
- "[Initiation of the trusted channel](#) " on page 143
- "[Termination of the trusted channel](#) " on page 143

- **portal_error_log and https-error_log log messages**

- "[Failure to establish a TLS session](#) " on page 143
- "[Initiation of the channel](#)" on page 144
- "[Termination of the channel](#)" on page 144
- "[Initiation of the trusted path](#) " on page 144
- "[Termination of the trusted path](#) " on page 144

- **Secure log messages**
 - "Initiation of self-test " on page 144
 - "Failure of self-test. Detected integrity violation" on page 145
- **upgrade.log log messages**
 - "Success of signature verification " on page 145
 - "Failure of signature verification " on page 145

MIFS log messages

Failure of key generation activity for authentication key

MDMPP40:FCS_CKM.1

2021-07-21 17:01:32,853 INFO [ApplicationLoggingHelper.log:57] (http-nio-127.0.0.1-8081-exec-10:[]) {pathInfo=/api/v2/configuration/SCEP/SCEP/issue_test_certificate} keyGenerator failed to generate key pair

Failure of the randomization process

MDMPP40:FCS_RBG_EXT.1

2021-07-21 16:00:02,838 INFO [ApplicationLoggingHelper.log:57] (MIRreportScheduler_Worker-3:[]) {} SecureRandom.nextBytes failed to generate random data, retry_count 1

Failure to establish a TLS session

PKGTL11:FCS_TLSC_EXT.1

2021-03-01 05:00:11,484 ERROR [TlsReportingSSLSocket.startHandshake:268] (pool-10-thread-1:[]) {} Failed to create the SSL socket. Reason TLS Peer Unverifiable: [java.security.cert.CertificateException: Could not find root certificate in the certificate chain presented by the server with subject: CN=corefcm.mobileiron.com, OU=EliteSSL, OU=Site Reliability Engineering, O="MobileIron, Inc.", STREET=415 E Middlefield Rd, L=Mountain View, ST=CA, OID.2.5.4.17=94043, C=US] Remote endpoint details: /34.206.49.117:443

Failure to verify presented identifier

2021-03-01 16:51:06,552 ERROR [SecurityModuleHostnameVerifier.verify:79] (http-bio-127.0.0.1-8081-exec-1144:[]) {pathInfo=null} Verification failed due to 'Certificate for <tl28-16x.exampIn't match any of the subject alternative names: [172.16.0.4, fc00:0:0:0:0:16:0:4, mismatched.dns.name, wrong-user@mismatched.dns.name]'

Establishment of a TLS session

2021-03-01 16:48:20,424 INFO [ApplicationLoggingHelper.log:94] (http-bio-127.0.0.1-8081-exec-1139:[]) {pathInfo=null} Created the TLS socket successfully. Remote endpoint details: 'tl28-16x.example.com/10.0.0.163:636' Local endpoint details: '/10.0.0.131:52450' using protocol/cipher: SUCCESS :TLSv1.2:TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256

Termination of a TLS session

```
2021-03-01 16:48:20,432 INFO [ApplicationLoggingHelper.log:82] (http-bio-127.0.0.1-8081-exec-1145:[]) {pathInfo=null} SSL Socket closed successfully. | Remote endpoint details: 't128-16x.example.com/10.0.0.163:636' Local endpoint details: '0.0.0.0/0.0.0.0:52444'
```

Failure to validate X.509 certificate – Expired Certificate

MDMPP40:FIA_X509_EXT.1(1)

```
2021-03-01 18:24:16,952 INFO [ApplicationLoggingHelper.log:82] (http-bio-127.0.0.1-8081-exec-1281:[]) {pathInfo=null} Error validating certificate path. Error message: Could not validate certificate: certificate expired on 20201106204500GMT+00:00. Exception stack trace:
org.bouncycastle.jcajce.provider.RFC3280CertPathUtilities.processCertA(Unknown Source)
org.bouncycastle.jcajce.provider.PKIXCertPathValidatorSpi.engineValidate(Unknown Source)
java.security.cert.CertPathValidator.validate(CertPathValidator.java:292)
com.mobileiron.security.CertPathUtils.validateCertPathRFC3280(CertPathUtils.java:445)
com.mobileiron.security.CertPathUtils.validateCertPathRFC3280UnknownCA(CertPathUtils.java:415)
com.mobileiron.security.CertPathUtils.buildAndvalidateCertPathRFC3280UnknownEE
(CertPathUtils.java:483)
com.mobileiron.security.CertPathUtils.getRootCertificateFromChain(CertPathUtils.java:502)
com.mobileiron.security.SecurityModuleTrustManager.addCertificatesToUntustedList
(SecurityModuleTrustManager.java:388)
com.mobileiron.security.SecurityModuleTrustManager.checkTrustedStrict
(SecurityModuleTrustManager.java:351)
com.mobileiron.security.SecurityModuleTrustManager.checkTrusted
(SecurityModuleTrustManager.java:249)
```

Failure to validate X.509 certificate – Revoked Certificate

```
2021-03-01 18:07:35,902 ERROR [TlsReportingSSLSocket.startHandshake:279] (http-bio-127.0.0.1-8081-exec-1252:[]) {pathInfo=null} Failed to create the SSL socket. Reason TLS Peer Unverifiable: [ java.security.cert.CertificateException: java.security.cert.CRLException: Unable to Verify Certificate 'CN=t128-16x.example.com, O=GSS, L=Catonsville, ST=MD, C=US' with serial number '186' Certificate CN=t128-16x.example.com, O=GSS, L=Catonsville, ST=MD, C=US serial number 186 status: revoked ] Remote endpoint details: t128-16x.example.com/10.0.0.163:636
```

Failure to validate X.509 certificate – Revocation Server no CRLsign

```
2021-03-01 18:02:43,514 INFO [ApplicationLoggingHelper.log:94] (http-bio-127.0.0.1-8081-exec-1227:[]) {pathInfo=null} Failed to retrieve the CRL for Certificate with Subject: CN=t128-16x.example.com, O=GSS, L=Catonsville, ST=MD, C=US Serial Number: 146 Issuer: CN=subsubca-unreachable-ecdsa, O=GSS, L=Catonsville, ST=MD, C=US
```

Failure to validate X.509 certificate –Missing BasicConstraints

```
2021-03-01 16:16:27,167 INFO [ApplicationLoggingHelper.log:82] (http-bio-127.0.0.1-8081-exec-1139:[]) {pathInfo=null} Failed to create the TLS socket. Reason TLS Peer Unverifiable: [ java.security.cert.CertificateException: Could not find root certificate in the certificate
```

chain presented by the server with subject: CN=tl28-16x.example.com, O=GSS, L=Catonsville, ST=MD, C=US] Remote endpoint details: tl28-16x.example.com/10.0.0.163:636

Failure to validate X.509 certificate – CA Flag is False

```
2021-05-13 12:42:09,348 ERROR [CertPathUtils.buildCertChain:683] (http-nio-127.0.0.1-8084-exec-7:[]) {pathInfo=/miclientcheckin} Found certificate [CN=subca-ca-flag-false-ecdsa, O=GSS, L=Catonsville, ST=MD, C=US with SN 13] that is not a valid CA, aborting building chain for [CN=371f7611b163e903]
```

Failure to validate X.509 certificate – No KeyCertSign KeyUsage

```
[Tue Aug 03 20:29:16.397257 2021] [ssl:debug] [pid 11051] ssl_engine_kernel.c(1360): [client 10.0.0.246:58920] AH02275: Certificate Verification, depth 0, CRL checking mode: none [subject: CN=636713576 / issuer: CN=subsubca-no-keyCertSign-ecdsa,O=GSS,L=Catonsville,ST=MD,C=US / serial: 7F / notbefore: Aug 2 20:29:11 2021 GMT / notafter: Aug 3 20:29:11 2022 GMT]
```

Failure to validate X.509 certificate – Bad Path Length

```
2021-08-03 12:24:01,897 INFO [ApplicationLoggingHelper.log:57] (http-nio-127.0.0.1-8084-exec-1:[]) {} EC : client : Certificate chain validation failed : Subject: CN=2045348614Serial Number106 Issuer CN=subsubca-issued-by-bad-path-length-ecdsa, O=GSS, L=Catonsville, ST=MD, C=US Berfore Mon Aug 02 12:23:56 UTC 2021 After Wed Aug 03 12:23:56 UTC 2022 |
```

```
-----BEGIN CERTIFICATE-----  
<BASE-64 Encoded Cert Removed For Brevity>
```

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

```
Subject: CN=subsubca-issued-by-bad-path-length-ecdsa, O=GSS, L=Catonsville, ST=MD, C=USSerial Number319 Issuer CN=subca-bad-path-length-ecdsa, O=GSS, L=Catonsville, ST=MD, C=US Berfore Tue Dec 01 20:51:06 UTC 2020 After Fri Nov 29 20:51:06 UTC 2030 |
```

```
-----BEGIN CERTIFICATE-----  
<BASE-64 Encoded Cert Removed For Brevity>
```

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

```
Subject: CN=subca-bad-path-length-ecdsa, O=GSS, L=Catonsville, ST=MD, C=USSerial Number311 Issuer CN=rootca-ecdsa, O=GSS, L=Catonsville, ST=MD, C=US Berfore Tue Dec 01 20:17:20 UTC 2020 After Fri Nov 29 20:17:20 UTC 2030 |
```

```
-----BEGIN CERTIFICATE-----  
<BASE-64 Encoded Cert Removed For Brevity>
```

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

```
Subject: CN=rootca-ecdsa, O=GSS, L=Catonsville, ST=MD, C=USSerial Number65537 Issuer CN=rootca-ecdsa, O=GSS, L=Catonsville, ST=MD, C=US Berfore Tue Oct 27 12:52:54 UTC 2020 After Thu Oct 27 12:52:54 UTC 2022 |
```

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

<BASE-64 Encoded Cert Removed For Brevity>

-----END CERTIFICATE-----

Reason: Certificate chain path Length error. Exception stack trace:

Failure to validate X.509 certificate – Explicit curve in certificate

9 19:14:25,557 INFO [ApplicationLoggingHelper.log:57] (http-nio-127.0.0.1-8084-exec-1:[]) {}
EC : client : Certificate chain validation failed :

Subject: CN=1978649938Serial Number106 Issuer CN=subsubca-explicit-ecdsa, O=GSS,
L=Catonsville, ST=MD, C=US Berfore Sun Aug 08 19:14:19 UTC 2021 After Tue Aug 09 19:14:19 UTC
2022 |

-----BEGIN CERTIFICATE-----

<BASE-64 Encoded Cert Removed For Brevity>

-----END CERTIFICATE-----

Subject: CN=subsubca-explicit-ecdsa, O=GSS, L=Catonsville, ST=MD, C=USSerial Number315 Issuer
CN=subca-ecdsa, O=GSS, L=Catonsville, ST=MD, C=US Berfore Tue Dec 01 20:51:05 UTC 2020 After
Fri Nov 29 20:51:05 UTC 2030 |

-----BEGIN CERTIFICATE-----

<BASE-64 Encoded Cert Removed For Brevity>

-----END CERTIFICATE-----

Subject: CN=subca-ecdsa, O=GSS, L=Catonsville, ST=MD, C=USSerial Number9 Issuer CN=rootca-
ecdsa, O=GSS, L=Catonsville, ST=MD, C=US Berfore Fri Nov 06 20:40:31 UTC 2020 After Mon Nov 04
20:40:31 UTC 2030 |

-----BEGIN CERTIFICATE-----

<BASE-64 Encoded Cert Removed For Brevity>

-----END CERTIFICATE-----

Subject: CN=rootca-ecdsa, O=GSS, L=Catonsville, ST=MD, C=USSerial Number65537 Issuer
CN=rootca-ecdsa, O=GSS, L=Catonsville, ST=MD, C=US Berfore Tue Oct 27 12:52:54 UTC 2020 After
Thu Oct 27 12:52:54 UTC 2022 |

-----BEGIN CERTIFICATE-----

<BASE-64 Encoded Cert Removed For Brevity>

-----END CERTIFICATE-----

Reason: Certificate must have named Elliptic Curves Exception stack trace:
com.mobileiron.security.SecurityModuleTrustManager.verifyNamedCurves
(SecurityModuleTrustManager.java:184)

Failure to establish connection to determine revocation status.

MDMPP40:FIA_X509_EXT.2

```
2021-03-01 18:02:43,514 INFO [ApplicationLoggingHelper.log:94] (http-bio-127.0.0.1-8081-exec-1227:[]) {pathInfo=null} Could not fetch CRL from: http://172.16.8.44/subsubca-unreachable-ecdsa.crl for certificate with Subject: CN=tl28-16x.example.com, O=GSS, L=Catonsville, ST=MD, C=US Serial Number: 146 Issuer: CN=subsubca-unreachable-ecdsa, O=GSS, L=Catonsville, ST=MD, C=US
```

Failed user login during enrollment

MDMPP40:FIA_ENR_EXT.1

```
2021-08-12 19:36:05,174 INFO [ApplicationLoggingHelper.log:57] (http-nio-127.0.0.1-8081-exec-8:[]) {pathInfo=/api/v2/Enrollment/CertReq.enroll} Registration failed for device with platform: Android. Reason: authentication failed for user gssuser - User: gssuser
```

Enrollment attempted after expiration of authentication data

MDMPP40:FMT_SAE_EXT.1

```
2021-05-19 20:35:36,556 WARN [DeviceRegistrationServiceImpl.getDeviceUUIDForValidRegistrationPin:2290] (http-nio-127.0.0.1-8081-exec-18:[]) {pathInfo=/api/v2/Enrollment/CertReq.enroll} Device ab0be7b7-08fc-43df-89ac-95b758754cb7: registration PIN expired  
2021-05-19 20:35:36,557 ERROR [MIDeviceServiceImpl.authenticateUserByPin:5179] (http-nio-127.0.0.1-8081-exec-18:[]) {pathInfo=/api/v2/Enrollment/CertReq.enroll} PinAuthFailed: Device not found for the PIN you entered. Check your PIN and try again.
```

Initiation of the trusted channel

MDMPP40:FTP_ITC.1(1)

```
2021-03-01 16:48:20,424 INFO [ApplicationLoggingHelper.log:94] (http-bio-127.0.0.1-8081-exec-1139:[]) {pathInfo=null} Created the TLS socket successfully. Remote endpoint details: 'tl28-16x.example.com/10.0.0.163:636' Local endpoint details: '/10.0.0.131:52450' using protocol/cipher: SUCCESS :TLSv1.2:TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
```

Termination of the trusted channel

```
2021-03-01 16:48:20,432 INFO [ApplicationLoggingHelper.log:82] (http-bio-127.0.0.1-8081-exec-1145:[]) {pathInfo=null} SSL Socket closed successfully. | Remote endpoint details: 'tl28-16x.example.com/10.0.0.163:636' Local endpoint details: '0.0.0.0/0.0.0.0:52444'
```

portal_error_log and https-error_log log messages

Failure to establish a TLS session

PKGTLS11:FCS_TLSS_EXT.1

General Structure of a TLS failure audit:

```
<<DATE_TIME_STAMP>> [ssl:info] [pid #####] SSL Library Error: error:1408E0F4:SSL routines:<<ERROR_MSG>>  
<<DATE_TIME_STAMP>> [ssl:info] [pid #####] [client <<IP ADDR>>:<<PORT #>>] AH01998: Connection closed to child ## with abortive shutdown (server <<ServerName>>:<<Port #>>)
```

Possible Error Messages are:

```
ssl3_get_message:unexpected message
SSL3_GET_RECORD:wrong version number
SSL3_GET_RECORD:block cipher pad is wrong
ssl3_get_finished:digest check failed
ssl3_get_client_hello:no shared cipher
ssl3_read_bytes:tlsv1 alert unknown ca (SSL alert number 48)
```

Sample Audit:

```
[Tue Mar 02 16:22:35.311412 2021] [ssl:info] [pid 7581] SSL Library Error: error:1408A0C1:SSL
routines:ssl3_get_client_hello:no shared cipher -- Too restrictive SSLCipherSuite or using DSA
server certificate?
```

```
[Tue Mar 02 16:22:35.311431 2021] [ssl:info] [pid 7581] [client 10.0.0.163:52092] AH01998:
Connection closed to child 5 with abortive shutdown (server micore11.gss.com:8443)
```

Initiation of the channel

MDMPP40:FPT_ITT.1(2) and MDMPP40:FTP_ITC.1(2)

```
[Mon Mar 01 20:28:48.798575 2021] [ssl:info] [pid 14679] [client 10.0.0.120:54639] AH01964:
Connection to child 7 established (server micore11.gss.com:443)
```

Termination of the channel

```
[Mon Mar 01 20:28:47.540264 2021] [ssl:debug] [pid 7170] ssl_engine_io.c(993): [client
10.0.0.120:54636] AH02001: Connection closed to child 4 with standard shutdown (server
micore11.gss.com:443)
```

Initiation of the trusted path

MDMPP40:FTP_TRP.1

```
[Mon Mar 01 20:27:23.614194 2021] [ssl:info] [pid 16036] [client 10.0.0.120:54592] AH01964:
Connection to child 2 established (server micore11.gss.com:8443)
```

Termination of the trusted path

```
[Mon Mar 01 20:27:23.711935 2021] [ssl:debug] [pid 16036] ssl_engine_io.c(993): [client
10.0.0.120:54592] AH02001: Connection closed to child 2 with standard shutdown (server
micore11.gss.c
```

Secure log messages

Initiation of self-test

MDMPP40:FPT_TST_EXT.1

```
<86>1 2021-07-23T20:06:05.732362+00:00 micore11c - - - RPM Verification Self-Test - Start
...
```



```

<86>1 2021-07-23T20:07:07.575773+00:00 micore11c - - - Verifying mobileiron-platform-base-
11.2.1.0-14.x86_64 RPM for integrity
<86>1 2021-07-23T20:07:07.893455+00:00 micore11c - - - RPM mobileiron-platform-base-
11.2.1.0-14.x86_64 has been verified
...
<86>1 2021-07-23T20:09:24.280314+00:00 micore11c - - - RPM Verification Self-Test - Complete

```

Failure of self-test. Detected integrity violation

```

<83>1 2021-07-23T20:22:47.916507+00:00 micore11c - - - /mi/tomcat/bin/commons-daemon.jar
File of mobileiron-core-mifs-11.2.1.0-14.noarch has failed RPM verification check
<83>1 2021-07-23T20:22:47.918503+00:00 micore11c - - - MD5 Sum has changed for
/mi/tomcat/bin/commons-daemon.jar
<86>1 2021-07-23T20:22:47.920735+00:00 micore11c - - - RPM Verification Self-Test - Failed

```

upgrade.log log messages

Success of signature verification

MDMPP40:FPT_TUD_EXT.1

```

2021-08-13 10:06:58 micore11c.gss.com gssadmin Checking Trusted signature for <<RPM Name>>
2021-08-13 10:06:58 micore11c.gss.com gssadmin Trusted Signature validation successful: <<RPM
Name>>

```

Failure of signature verification

MDMPP40:FPT_TUD_EXT.1

```

2021-08-13 08:14:10 micore11c.gss.com gssadmin ++++++ Initializing upgrade:
mobileiron-11.2.1.0-29 ++++++
2021-08-13 08:14:10 micore11c.gss.com gssadmin ++++++ Initializing download:
mobileiron-11.2.1.0-29 ++++++
Loaded plugins: changelog, list-data, priorities, ps, remove-with-leaves, show-
: leaves, verify
Repository mi-base is listed more than once in the configuration
Repository mi-updates is listed more than once in the configuration
Resolving Dependencies
--> Running transaction check
---> Package mi-buildinfo.noarch 0:11.2.1.0-26 will be updated
<<RPM List removed for brevity>>
--> Finished Dependency Resolution
Dependencies Resolved

```

```

=====
Package                               Arch      Version      Repository

```

