Application Management for Android

VMware Workspace ONE UEM 2109

vmware®

You can find the most up-to-date technical documentation on the VMware website at:

https://docs.vmware.com/

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2021 VMware, Inc. All rights reserved. Copyright and trademark information.

Contents

Managing Android Applications 1

Use the Workspace ONE UEM powered by AirWatch to push Android public applications, internal applications, and web apps to Android devices.

The managed Google Play Store is the recommended way to manage all your application deployment use-cases for Android devices. Managed Google Play loads in an iframe within the Workspace ONE UEMconsole whenever a public application is added and when an Android Enterprise EMM registration is configured. The iframe is opened through the API integration with Google Play and is not hosted by VMware. Applications that you push through the integration of Workspace ONE UEM and Android have the same functionality as their counterparts from the Google Play Store.You can also host the application .apk file as a local file. This option applies to Android 6.0+ devices only.

Internal apps are company-specific apps developed by your organization that are available for users to access from their device but not searchable in the public app store. You can add internal apps to Google Play as a private application. Although these private apps are managed as public apps and are available for assigned users to access, they are not searchable in the public app store.

You can use Workspace ONE UEM features to apply policies to the applications. For example, you can add configurations that make using the application more convenient and you can configure settings that make using the application more secure.

- To add convenience of use, configure the Send Application Configuration option. Application configurations allow you to pre-configure supported key-value pairs and to push them down to devices with the application. Examples of supported values may include user names, passwords, and VPN settings. Support value depends upon the application.
- To add secure features, use Workspace ONE UEM profiles for Android. Profiles let you set passcodes, apply restrictions, and use certificates for authentication.

The Workspace ONE UEM console allows you to push alpha, beta, or production versions of apps. Using alpha and beta versions of apps allows for testing for compatibility and stability before pushing the production version. You can select specific smart groups for testing and use flexible deployment to determine which users receive which version of the app. If you don't select whether to push the alpha or beta version, the production version is automatically assigned.

Important VMware productivity apps (Browser, Boxer, Content Locker, etc) are not supported with Android (Legacy) Knox container deployments, such as Dual Persona or Container Only Mode, due to technical limitations with Knox container data separation. The Workspace ONE Intelligent Hub manages the container from the outside, and is not able to communicate with apps on the inside. Since the apps require a direct link to the Workspace ONE Intelligent Hub in order to communicate with the Workspace ONE UEM console, the apps cannot be configured inside the container. In order to use productivity apps with Knox, the device must be enrolled using Android Enterprise on a device running Knox 3.x or higher.

Workspace ONE Intelligent Hub for Android

Workspace ONE Intelligent Hub for Android is an application that enables the Native Android SDK API layer of management to which Workspace ONE UEM connects. Workspace ONE UEM engages Native Android SDK APIs on Android devices for management and tracking capabilities. are available to any third-party application, including the Workspace ONE Intelligent Hub and any other application using the AirWatch Software Development Kit (SDK).

With the AirWatch SDK, applications can take advantage of key MDM features that are available such as:

- Compromised Device Detection
- GPS Tracking
- Additional Telecom Detail
- Additional Network Details such as IP address
- Additional Battery and Memory statistics
- Native number

After enrolling, use the Workspace ONE Intelligent Hub to access and manage device information and settings. Access device information from the following tabs on the left of the device display:

- **This Device** – Displays the name of the enrolled end user, the device-friendly Name, current enrollment status, connectivity method, and compliance status.
- **Device Status** – Displays the current enrollment status including:
  - The server to which the device is connected.
  - The organization group to which the device is enrolled.
  - The current network status including the active Wi-Fi SSID to which the device is connected.

- **Compliance** – Displays a list of compliance policies currently active for the device.
- **Profiles** – Displays a list of profiles currently installed on the device. From the profiles list, you can refresh and reapply profiles from your device that might be out of sync or uninstalled.
- **Managed Apps** – Displays a list of apps managed by Workspace ONE UEM installed on the device and their install status.
- **About** – Displays the version number of the Workspace ONE Intelligent Hub installed on the device and provides a hyperlink to the associated Privacy Policy agreed to upon device enrollment.

Perform basic device management functions from the Workspace ONE Intelligent Hub menu at the top of the display:

- **Sync Device** – Sync latest device information and receive updates from IT admin.
- **App Catalog** – Launch the application catalog within the Workspace ONE Intelligent Hub or the native web browser, if applicable.

Additional functionality is accessible from the application menu in the upper-right corner of the display:

- **Edit Phone Number** – Modify the assigned phone number, if applicable.
- **Send Debug Log** – Transmit a debug log for the device to Workspace ONE UEM.
- **Remove Device** – Unenroll the device from Workspace ONE UEM.

Android devices running Android 6.0 (Marshmallow) and above use the power saving options for idle apps and devices. If a user unplugs a device and leaves it stationary, with its screen off, for a period, the device goes into **Doze** mode, where it attempts to keep the device in a sleep state. There will be no network activity during this time. Doze mode affects how the Workspace ONE Intelligent Hub reports information back to Workspace ONE UEM.

When a device is on battery power, and the screen has been off for a certain time, the device enters Doze mode and applies a subset of restrictions that shut off app network access and defer jobs and syncs. After a device is in doze mode for a period, the system sends the remaining Doze restrictions to wake locks, alarms, GPS, and Wi-Fi settings.

Additionally, mode allows the device to determine that an app is idle when the user is not actively using it. When devices are in either state, the Workspace ONE UEM console will not receive reports on device details. When the user plugs a device in to charge or opens an app, the device can resume normal operations and reporting from AirWatch apps installed on the device to the Workspace ONE UEM console resumes.

The Hub and SDK-Built Applications

AirWatch offers an SDK to integrate into applications you build for the Android platform. Integrating the SDK into your applications enables the application to use AirWatch features. These features include controlling authentication to SDK-built applications and sharing a single-sign on session between applications that use the SDK.

However, you must enable **Key Encryption with User Input** so that the Workspace ONE Intelligent Hub can care share an application passcode or an SSO session with other SDK applications.

For information on the AirWatch SDK for Android, see AirWatch SDK for Android documentation.

For information on SDK features in the Workspace ONE UEM console , see MAM Features With SDK Functions documentation.

For information on the option **Key Encryption with User Input**, see Devices & Users / Android / Security in the Workspace ONE UEM System Settings documentation.

Application Types and Supported OS Versions for Android

Workspace ONE UEMclassifies applications as native (internal, public, purchased), SaaS, and Web. You upload applications depending on the type. Workspace ONE UEM supports the following OS Versions for Android applications based on the application type.

Table 1-1. Application Types and Supported OS Versions for Android

Application Type	Supported Platforms
Internal	Android v4.4+
Public (Free and Paid)	Android v4.4+
Web Links	Android v4.4+
SaaS	Android v4.4+

This chapter includes the following topics:

Deploying Internal Application on Android Devices

Internal apps are company-specific apps developed by your organization that you might not necessarily want to be searchable in the public app store, but you want your users to have access to this application from their device.

There are two options for deploying internal applications:

- Add it to Google Play as a private application. These applications are added as public applications in the Workspace ONE UEM console after publishing in Google Play.
- Host the application .apk file as a local file. For Android 6.0+ devices only.

If you are deploying internal apps on Android Work profile devices, add internal apps to Managed Google Play Store so that they are available to the Android-specific users. Upload your application by logging into the Google Play Developer Console with your enterprise credentials. There is an option to enable, Restrict Distribution, which only allows users of your domain to view this application on Managed Google Play Store (the badged play store). Once you have added your internal application to the developer console, these apps are treated as public applications.

Note There are a few changes to Corporate Owned Personally Enabled (COPE) in Android 11. For more information, see Changes to Corporate Owned Personally Enabled (COPE) in Android 11.

- Internal applications that are hosted by Workspace ONE UEM can no longer be pushed on the personal side of the device. Both internal apps that are pushed as private apps and public apps must be deployed to the work profile only.
- Any other functionality such as Compliance Rules that rely on the internal application is no longer supported.

Add Assignments and Exclusions to your Android Applications

Adding assignments and exlusions provides you flexible deployment process and let's you schedule multiple deployment scenarios for a single application. After you approve the application from the Google Play Store, you will be redirected to the Workspace ONE UEM console to assign the applications to smart groups on the assignment tab. You can add a single assignment or multiple assignments to control your application deployment and prioritize the importance of the assignment by moving its place in the list up for most important or down for least important. Also,you can also exclude groups from receiving the assignment.

1 Navigate to **Resources** > **Apps** > **Native** > **Internal** or **Public**.

2 Upload an application and select **Save & Assign** or select the application and select **Assign** from the actions menu.

3 On the **Assignments** tab, select **Add Assignment** and complete the following options.

- In the **Distribution** tab, enter the following information:
-

Setting	Description
Name	Enter the assignment name.
Description	Enter the assignment description.
Assignment Groups	Enter a smart group name to select the groups of devices to receive the assignment.

Setting	Description
Deployment Begins On	**Deployment Begins On** is available only for internal applications.Set a day of the month and a time of day for the deployment to start. For successful deployment consider traffic patterns of your network before you set a beginning date with bandwidth.
App Delivery Method	■ **On Demand** – Deploys content to a catalog or other deployment agent and lets the device user decide if and when to install the content. This option is the best choice for content that is not critical to the organization. Allowing users to download the content when they want helps conserve the bandwidth and limits unnecessary traffic. ■ **Automatic** – Deploys content to a catalog or other deployment Hub on a device upon enrollment. After the device enrolls, the system prompts users to install the content on their devices. This option is the best choice for content that is critical to your organization and its mobile users

- In the **Restrictions** tab, enter the following information:

Settings	Descrption
Managed Access	Enable adaptive management to set Workspace ONE UEM to manage the device so that the device can access the application. Workspace ONE controls this feature and is not supported by the AirWatch Catalog.

- In the **Tunnel** tab, enter the following information:

Setting	Description
Android Legacy	Select the **Per-App VPN Profile** you like to use for the application and configure a VPN at the application level.

4 Select **Create**.

5 Select **Add Assignment** to add new app assignments for your application.

6 Configure flexible deployment settings for your application by editing the schedules and priority for your deployments. Options that are displayed on this window are platform-specific.

a

Setting	Description
Copy	From the ellipses-vertical, you can click copy if you choose to duplicate the assignment configurations.
Delete	From the ellipses-vertical, you can delete to remove the selected assignment from the application deployment.

Setting	Description
Priority	You can modify the priority of the assignment you configured from the drop-down menu while placing the selected assignment in the list of assignments. Priority 0 is the most important assignment and takes precedence over all other deployments. Your devices receive all the restrictions distribution policies and the app configuration policies from the assignment group which has the highest priority. If a device belongs to more than one smart group and you assign these smart groups to an application with several flexible deployments, the device receives the scheduled flexible deployment with the most immediate **Priority**. As you assign smart groups to flexible deployments, remember that a single device can belong to more than one smart group. In turn, one device can be assigned to more than one flexible deployment for the same application. For example, if Device 01 belongs to Smart Group HR and Smart Group Training. You configure and assign two flexible deployments for application X, which include both Smart Groups. Device 01 now has two assignments for application X. ■ Priority 0 = Smart Group HR, to deploy in 10 days with On Demand ■ Priority 1 = Smart Group Training, to deploy now with Auto Device 01 receives the priority 0 assignment and gets the application in 10 days because of the assignments priority rating. Device 01 does not receive the priority 1 assignment.
Assignment Name	View the assignment name.
Description	View the assignment description.
Smart Groups	View the assigned smart group.
App Delivery Method	View how the application pushes to devices. **Auto** pushes immediately through the AirWatch Catalog with no user interaction. **On Demand** pushes to devices when the user initiates an installation from a catalog.
EMM Managed Access	View whether the application has adaptive management enabled.

7 Select the **Exclusions** tab and enter smart groups, organization groups, and user groups to exclude from receiving this application.

- The system applies exclusions from application assignments at the application level.
- Consider the organization group (OG) hierarchy when adding exclusions. Exclusions at a parent OG do not apply to the devices at the child OG. Exclusions at a child OG do not apply to the devices at the parent OG. Add exclusions at the desired OG.

8 Select **Save & Publish**.

Deploy Application on your Android Devices through Managed Google Play Store

The managed Google Play Store is the recommended way to manage all your application deployment for Android devices. Managed Google Play loads with the Google play iframe in the Workspace ONE UEM console whenever a public application is added and when an Android

Enterprise EMM Registration is configured. The Google play iframe opens through the API integration with Google Play and is not hosted by VMware.

Deploy Public Applications through Managed Google Play Store

Search the Google Play Store directly from the console to add applications to the Managed Google Play Store for your users.

1 Navigate to **Resources** > **Apps** > **Public** > **Add Application**.

2 Select **Android** from the **Platform** drop-down menu.

3 Select one of the following options to add an application:

Setting	Description
Search App Store	Select to search for the application in the app store. Google Play launches within the console through an iframe.
Enter URL	Enter the URL of the app.
Import From Play	Select to import previously approved applications.

4 Select **Next** or enter the **Name** of the applications you want to add to the integration. Google Play can open directly from the console.

5 Find desired apps by using the **Search** text box or browsing through the apps section.

6 Review the permissions the application requires on the device, and select **Approve**.

7 Future updates to the application may require further permissions on the device. If you choose to approve the updates automatically and allow them to be pushed to devices, consider selecting **Keep approved when app requests new permissions**.If an application is updated, ensure it does not need to get approved in the Google Play Store.

8 Configure options on the **Details** tab.

Setting	Description
Name	View the name of the application.
View in App Store	View the store record for the application where you can download it and get information about it.
Categories	Use categories to identify the use of the application. You can configure custom application categories or keep the application's pre-coded category.
Supported Models	Select all the device models that you want to run this application.
Is App Restricted to Silent Install Android	Assign this application to those Android devices that support the Android silent uninstallation feature. However, you can control what applications you push to your Android standard devices or your Android enterprise devices. Android enterprise devices support silent activity.
Managed By	View the organization group (OG) that the application belongs to in your OG hierarchy.

9 (Optional) Assign a Required Terms of Use for the application on the **Terms of Use** tab.

Terms of use state specifically how to use the application. They make expectations clear to end users. When the application pushes to devices, users view the terms of use page that they must accept to use the application. If users do not accept the terms of use, they cannot access the application.

10 Select the **SDK** tab and assign the default or custom **SDK Profile** and an **Application Profile** to the application. SDK profiles apply advanced application management features to applications.

11 Select **Save & Assign** to configure flexible deployment options for the application.

Deploy a new Private Application through Managed Google Play Store

You can publish applications developed by your organization or the applications that are developed for your organization can be hosted and distributed through the Managed Play Store. While adding a public app on an organization group with Android Enterprise enabled, the iframe is loaded and the private apps are available in the left menu. Additional information such as a description, images, and more can be added in the Advanced options after uploading. These applications are visible to devices managed by the enterprise, and they aren't public outside the enterprise.

Note Private apps uploaded through the iframe can never be visible in the public Google Play Store. If the application may eventually be made public outside your organization, it is recommended to publish the application directly in the Google Play Developer Console instead of using the iframe.

Before you begin:

- Make sure that your Workspace ONE environment is registered to Android Enterprise Mobility Management (EMM).
- The **APK** file with the same application ID is not published in the Android public play store.
- Google publisher account have a limit of 15 APKs per day. Which means, you can only upload up to 15 applications per day.

1 Navigate to **Resources** > **Apps** > **Public** > **Add Application**.

2 Select **Android** from the **Platform** drop-down menu. Leave the **Name** blank and select **Next**Google Play console opens directly from the console.

3 Access the **Private Apps** from the left menu.

4 Click the “+” icon to add a new application and select **Upload APK**.

Note

- Uploading through the iframe publishes the application in as little as 10 minutes and waives the one-time fee that is charged to create a Google Developer account.
- Private applications can never be uploaded more than once as the Google Play ensures that each of the application has a unique package name.
- Deleted Private applications cannot be reuploaded with the same package name. Delete the private applications only if you never want to use the same package name again. The package name is a unique name to identify a specific app.

Application is displayed in the **Private app** section and a notification may take up to 10 minutes. You can now close this screen. The application is listed under the Public apps.

5 (Optional) You can edit the logo that is displayed in the UEM console by using the pencil icon beside the application.

6 Select **Assign** and **Add Assignment**.

7 Select the organization group or the smart group you like to assign the app to and click **Add**.

8 In the **Update Assignment** window, click **Save and Publish** to confirm.

9 **Publish** the app assignment.If the deployment is set to **Automatic**, the application gets installed automatically on the device and is displayed in both the console and your device.

Deploy Custom Testing Tracks before releasing the Production Version

Sometimes you may want to test your application and fix any technical or user experience difficulties with minimal user impact, and later select to release the best version of your application to most users. console provides you with the ability to test and deploy any number of custom releases before releasing the production version.

Supporting custom release of the application lets organizations test third-party apps, and any private apps they develop in-house. You can publish applications to the custom testing tracks in the Google Play console and then assign the applications to smart groups.

Before you begin:

- Make sure you have the APK file for the new version you want to publish.
- If there are multiple devices registered to one user but assigned to different tracks, navigate to **Groups and Settings** > **Devices & users** > **Android** > **Android EMM Registration** > **Enrollment Settings** and set the **Work Managed Enrollment Type** as **Device-Based**. The setting ensures that a different GoogleID record is generated per device, and so different app versions from the Managed Play console can be assigned.

1 Navigate to **Resources** > **Apps** > **Public** > **Add Application**.

2 Select **Android** from the **Platform** drop-down menu. Leave the **Name** blank and select **Next**.Google Play console opens directly from the console.

3 Access the **Private Apps** from the left menu.

4 Select the Private application for which you want to add the custom release for testing.

5 Click **Make advanced edits** under Advanced editing options.

6 You are directed to the Google Play console login page. In the Google Play console, complete the following steps to create **Closed Test Track** of the application.

   a Log in to the Google Play console using the google account tied to your Workspace ONE tenant. Go to your app and navigate to **Release management** > **App release**. Select **Create Closed Track**.

   b Under **Organizations**, click **Edit**.

   c Select the organization corresponding to the Workspace ONE organization group and click **Done**.

   d Click **Create Release**.

   e Add the APK file.After adding the APK file, you can see details about the version code and size of the file.

   f Click **Save** at the bottom of the screen, then **Review**.View any of the warning messages and make necessary changes to the app, as requested.

   Note It is safe to ignore a warning message about the testers, as it is defined though Workspace ONE later.

   g Click **Start Rollout** and **Confirm** the rollout.

7 In console, select the application from **Resources** > **Apps** > **Native**.

8 Click **Assign** and **Add Assignment**.

9 Select the **Assignment Group** for the new custom release of your application.

10 In the **Distribution** tab, select the closed testing track that you have created from the **Pre-release Version** drop-down as per your deployment.

11 Click **Save and Publish**.

12 Click **Publish** to confirm the assignment. The version corresponding to the rules in step 11a is made available to appropriate groups.

Deploy Private Applications to multiple servers

You can create private applications for your organization and deploy them to multiple servers.

To publish private applications from the Google Play Console, you must register for a Google Play developer account. The account gives you the correct administrator privileges to upload and publish private applications to managed Google Play. You can then use console to distribute these applications to users.

1 In your console server#1, navigate to **Resources** > **Apps** > **Public** > **Add Application**.

2 Select **Android** from the **Platform** drop-down menu. Leave the **Name** blank and select **Next**. Google Play console opens directly from the console.

3 Access the **Private Apps** from the left menu.

4 Select the private application for which you want to add multiple servers.

5 Click **Make advanced edits** under **Advanced editing options**.

   a Sign in to the Google Play Console.

   b Go to **Pricing & Distribution** > **User programs** > **Managed Google Play**.

   c Click **Choose Organizations**.

   d For each organization that you want to publish the application to, enter the **Organization ID**.

   e To get the Organization ID of your Workspace ONE server, complete the following steps:

      1 Sign in to the Managed Google Play store using the Google account that is associated with the server#2 (Android EMM Registration instance) for which you want to make the private app available.

      2 Click **Admin Settings**.

      3 Copy the **Organization ID** string from the Organization information box.

   a Paste the Organization ID, add a description (or name) and click **Add**.

   b Click **Done**.

When you are ready to publish your application, you can either create and rollout a production release or an alpha/beta track. After your application is published, you can create releases or set up a staged rollout.

Deploy Web Applications through Managed Google Play Store

Web applications are shortcuts on android devices that the users can open to navigate to the pre-defined URLs. They are installed in a silent mode. Web applications can be managed on the android devices similarly to public applications. Web apps requires Google Chrome to function and is managed as a public application. To do so, administrators need to set the title, URL, display mode, and the icon. The managed Google Play store loads in an iframe that creates a Web App object that is treated by the Google Play, and the Android OS as if it were a Public application.

Note You can create up to 15 new web apps per day.

1 Navigate to **Resources** > **Apps** > **Public** > **Add Application**.

2 Select **Android** from the **Platform** drop-down menu.

3 Select **Search App Store** to search for the application in the app store. Leave the Name blank and select **Next**. Google Play opens directly from the console.

4 Access the **Web Apps** from the left menu.

5 Create a **Web App**.

   a Enter the **Title** and **URL**.

   b Select the **Display Mode**.

   c Upload the **Icon**.

   d Select **Create**.

   e After you **Save** the Web App, select the **Back** arrow at the top-left of the screen.

   Note Approximately it takes close to 30 minutes for the app to be available. Once the app is available, it takes close to 10 minutes to publish a web application.

6 Select the **Web App**.

7 Choose the **Select** option at the bottom of the screen.

8 Select **Save & Assign** to configure flexible deployment options for the Web App.

   Note Publishing the web application creates the application. However, publishing to users does not happen until you assign the application within the console.

Add Web Links for Android Devices from the Workspace ONE UEM console

Web links applications function much like an application on a device. Web Links are also known as shortcuts. They provide end users a way to access a URL directly from an icon on the menu of their device. The end user sees the web links application icon and title, selects the application, and connects directly to a specified URL. Web links applications are useful for navigation to extended URLs with many characters. They require manual user approval to add the short-cut to the home screen. You can place web links application icons on the springboard. These icons connect end users to internal content repositories or login screens, so end users do not open a browser and enter a long URL. Web Links can use a custom URI to open specific browsers and is managed as a profile, rather than Web Apps which must use the Chrome Browser.

You can add web links as an application in the **Resources** section of the Workspace ONE UEM console. For more information, see Add Web Links Applications.

Organizing your Applications in the Managed Play Store

Administrators can simplify access to recommended applications by adding applications into collections which are displayed as rows on the managed Play Store. After enabling this feature, there is a minimum requirement of one collection at all times. Apps which have not been assigned to a collection can only be found in the managed Play Store using the search functionality.

Once an environment has begun using collections, the managed Google Play cannot be reverted to an earlier state. Because the change to collections cannot be rolled back, customers are highly encouraged to test the feature in a sandbox environment to ensure it aligns with the desired end-user experience and functionality before rolling it out to any production environments.

1 Navigate to **Resources** > **Apps** > **Public** > **Add Application**.

2 Select **Android** from the **Platform** drop-down menu.

3 Select **Search App Store** to search for the application in the app store. Leave the **Name** blank and select **Next**. Google Play opens directly from the console.

4 Access the **Organize Apps** from the left menu.

5 Create collections and add apps to your collection to set the Play Store layout.

Configure Samsung Native Email in the Workspace ONE UEM console

Samsung Native Email enables users to manage multiple personal and business email accounts seamlessly. Samsung Native Exchange email is configurable within Android Fully managed, Work Profile, and Fully managed device with a work profile (previously COPE), enrollment modes using Application Configurations. You can configure Samsung Native email with or without certificate based authentication.

Complete the following steps to configure Samsung Native Email settings on the UEM console.

Prerequisites

- If you are using Certificate based authentication, be sure to create a Credentials profile prior to setting up app configuration. For more information, see Deploy Credentials.
- The certificate(s) must be created and installed on the device, either via a Credentials Profile or manual install of certificates, before the app configuration is delivered.
- You must know the alias of the certificate(s) or use a lookup variable for the alias.

Note To prevent the email configuration from failing:

- In the Certificate Request Template, use a Lookup Value to determine the certificate Subject Name. This can be used for the alias.
- In the App Configuration for Samsung Email, select the same Lookup Value as entered above for the necessary certificate settings.

Procedure

1 Navigate to **Resources > Apps > Public > Add Application** .

2 Select Android from the Platform drop-down menu.

3 Select **Search App Store** from the Source text box.

The Google Play Store opens directly from the Workspace ONE UEM console.

4 Select the Samsung Email app and then click Approve.

5 Select **Save & Assign** to continue, then select Add Assignment.

6 Scroll down to Application Configuration and select Enabled to view and configure Exchange or Email settings.

7 Use lookup values to configure dynamic options, such as user name, email address, or even certificate aliases.

OEMConfig on Android Enterprise Devices

OEMConfig is a standard solution for Android original equipment manufacturers (OEM) to provide additional management capabilities to administrators, on top of what is natively offered by the Android Enterprise. OEMConfig is an application that is built and maintained by the OEM and hosted on Google Play. The application takes advantage of AppConfig standards by allowing the administrator to dynamically configure any setting desired that the OEM offers in a data-driven user interface. Because the settings are data-driven and app-based, console upgrades are not required to access the latest settings offered by the OEM.

Use OEMConfig applications to add, create, and customize OEM specific settings for Android Enterprise devices. The application is published to devices through UEM and silently installed using Android Enterprise Managed Google Play. Customized settings are delivered to the application during or post-install, and the application calls the corresponding, proprietary APIs on the device. Different OEM's can include different settings, and these settings can vary depending on the management mode (Work Managed, Work Profile, or COPE). The available settings depend on what the OEM includes in their OEMConfig app. Contact your OEM vendor for more information on their support of OEMConfig.

Configure OEM Settings in the UEM console

OEMConfig is typically used to configure settings that are not built in to UEM console. Different original equipment manufacturers (OEM) include different settings. The available settings depend on what the OEM includes in their OEMConfig application.

Complete the following steps to configure the OEM settings for an OEMConfig application in the UEM console.

Before you start configuring OEMConfig on your devices, consider the following caveats:

- OEM settings is a data-driven user interface that uses text boxes and support lookup values for the user or device-specific configurations.

- If any of the OEM settings is left blank, or is not selected, UEM console does not send the key-value pair to the device, and it is excluded from the configuration.
- **Clear** button clears all the values from the current configuration, including the default values.
- Use the **Clear** button to set a subset of configurations.
- Use the **Clear** button on each of the configuration page to clear the configuration settings.
- An OEMConfig app is built by the OEM, and uploaded to Google Play. If it is not on Google Play, contact the OEM for more information.
- You are on UEM console 1907 or later.

1 Get the OEMConfig app from the Managed Google Play Store.

   a Navigate to **Resources** > **Apps** > **Public** > **Add Application**.

   b Select **Android** from the **Platform** drop-down menu.

2 In the pop up, fill out the following text boxes with the supplied information:

Setting	Description
Managed by	Select the Organization Group that you set up to manage applications. Only IT administrators that belong to that Group can edit OEMConfig application configurations.
Platform	Android
Source	Search App Store
Name	OEMConfig App. For example, enter **`Knox Service Plugin`** if you are trying to configure OEM settings for the Knox Service Plugin.

3 Click **Approve** to add the OEMConfig application as an approved application.

4 Edit the **App Configurations** to activate or deactivate the policies. From the **Assignments** tab, select Add Assignment.

   a In the detailed view page, click **Assign**.

   b Select your OEMConfig App. Click **Edit**.

   c **Edit** the Application Configuration to configure the OEM settings for the OEMConfig application.

5 In the app configuration screen, configure the OEM settings and modify the policies for your deployment.

6 On the Update Assignment pop-up window, click **Save and Publish**.

7 On the Preview Assigned Devices pop-up window, click **Publish**.

Retrieve Feedback from OEMConfig Applications

Google offers a channel that allows applications to send configuration feedback to the UEM console console for various reasons. However, the most common use case for sending feedback on managed configurations is for a quick detection of errors. The feedback can alert IT admin to errors and help them take action to correct a problem. UEM console lets you view feedback reported across all devices and supported applications.

Complete the following steps to view the feedback reported across all the devices and supported applications and to force update feedback for a specific device to get the latest update.

Before you begin:

- Workspace ONE Intelligence handles the retrieval and storage of the feedback data. To view the application feedback in the UEM console , you must opt for Workspace ONE Intelligence To access Workspace ONE Intelligence , navigate to **Hub** > **Intelligence**, select **Opt-in**, and select **Launch** after installing the Workspace ONE Intelligence Connector service. Workspace ONE Intelligence directly requests the feedback from Google through the API calls.

  Note In the case that Android Managed App Configurations Feedback Channel is in use, slow loading times for the **Application Details** > **Devices page**. For more information, see Long Load Times on Android Application Details Devices Page.

- Workspace ONE Intelligence feedback service can be turned on only for customer level organization groups.

  Complete the following steps to configure OEM settings in the UEM console:

  a You have two ways of viewing the feedback reported across all the devices and supported applications:

    1 You can navigate to **Resources** > **Apps** > **Native** > **Public** and click the OEMConfig application. In the Details view, select **Devices** and under **Assigned Configuration**, select **View Feedback**.

    2 You can go to **Devices** > **Details View** > **Apps** and select **View Feedback**.

  You can view the feedback reported for the application for quick detection of errors.

  b In the Feedback screen, you can perform the following actions:

    1 Click **Reload** to refresh the latest data from Workspace ONE Intelligence.

    2 Click **Edit Configuration** to go to the device configuration screen to modify and fix configuration errors.